International Master's Thesis

Development and integration of a control system for flexible grippers

Nikolay Kramarev
Technology

Studies from the Department of Technology at Örebro University
Örebro 2010

Development and integration of a control system for flexible grippers

Studies from the Department of Technology
at Örebro University

Nikolay Kramarev

Development and integration of a control system for flexible grippers

Supervisor: **professor Ivan Kalaykov**

© Nikolay Kramarev, 2009

Title: Development and integration of a control system for flexible grippers

ISSN 1404-7225

Abstract

Various robotic grippers support or even replace human beings on particular tasks, for example, different industrial applications. However, in some cases complex tasks are required and specifications of the industrial gripper should be more advanced. The adjustment for dynamic environment should be done automatically without a human intervention in a control process.

The aim of this master's project was to develop a control software system for given mechanical gripper prototype developed in the AASS research lab. Gripper control program was implemented by using Galil motion control hardware and Galil specific low-level programming language. Gripper was equipped with tactile sensors on fingertips to adjust a grasping behavior for objects with different shapes and sizes.

In this thesis the control system for gripper device as well as its integration into the ¨intelligent arm˝system was implemented. The ¨intelligent arm˝system is an ABB IRB 140 industrial robotic manipulator with IRC5 controller and motion capture system installed in the AASS research lab. Communication between Galil and IRC5 controllers allowed the gripper to be managed by IRC5. On the other hand, gripper could communicate its condition into IRC5 and therefore affect the arm motions.

Acknowledgements

First, I want to thank my supervisor professor Ivan Kalaykov for his support and guidance during my work on this project. A big thank to all who helped to prepare a practical part of the project and especially to Anany Ananiev, Boyko Iliev and Alexander Skoglund. And of course thanks to all who make my education process fascinating and life enjoyable during years at the Örebro University.

I would like to thank my wife, Olesya, my parents and whole my family for supporting me in all ways during my study. Without all of them a lot of thing in my life would not be possible.

Contents

1 Introduction **15**
- 1.1 Problem . . . 15
- 1.2 Motivation . . . 17
- 1.3 Objectives . . . 18
- 1.4 Contributions . . . 19
- 1.5 Thesis structure . . . 20

2 Mechanical gripper prototype and control hardware tools **23**
- 2.1 Flexible gripper mechanics . . . 23
- 2.2 Galil motion control system . . . 29
  - 2.2.1 DMC-2143 controller . . . 30
  - 2.2.2 AMP-20540 amplifier . . . 31
  - 2.2.3 Theory of operation . . . 32
- 2.3 Tactile sensors system . . . 35
- 2.4 The ¨intelligent arm˝system . . . 37
- 2.5 Vice gripper . . . 38

3 Vision guidance system for grasping by demonstration **41**
- 3.1 Automatic grasping based on demonstration . . . 41
- 3.2 Objects exploration and features extraction . . . 42
  - 3.2.1 Motion capture System . . . 42
  - 3.2.2 Grasp-related features extraction . . . 43
- 3.3 Flexible gripper control program . . . 43

4 Implementation **45**
- 4.1 Hardware . . . 45
  - 4.1.1 Flexible gripper wiring . . . 45
  - 4.1.2 Flexible gripper hardware integration . . . 46
- 4.2 Software . . . 48
  - 4.2.1 Flexible gripper control program . . . 49
  - 4.2.2 Vice gripper Galil control program . . . 52
  - 4.2.3 Tactile sensors system module . . . 54

4.2.4 Main control program 56

5 Experimental Results **59**
5.1 Integrated test platform content 59
5.1.1 Integrated test platform description 62

6 Conclusion **67**

List of Figures

1.1 The integrated system modules 16
1.2 The integrated system flow 17
1.3 Relation of grippers flexibility vs expenses 18

2.1 FG configuration (Top view) 24
2.2 FG prototype 24
2.3 Three grasp points configurations 25
2.4 Brushless DC motor components 26
2.5 Three phase brushless DC motor mechanism 26
2.6 The schematic diagram of the connection of three phase brushless DC motor to the controller 27
2.7 Schematic diagram of Inductive Proximity sensor 28
2.8 Inductive Proximity sensor 28
2.9 Mechanical limit switch 29
2.10 Voltage level adjustment circuit 29
2.12 DMC-2143 functional elements 30
2.11 Elements of motion control system 30
2.13 AMP-20540 amplifier 31
2.14 Elements of Servo System 32
2.15 Functional elements of motion control system 33
2.16 Tactile sensors system 35
2.17 Table of DSACON32-H parameters 36
2.18 DSACON 32-H controller 36
2.19 DSA 9205 tactile transducers 37
2.20 Components of the “Intelligent arm system” 38
2.21 Vice gripper degrees of freedom 39
2.22 Force-sensing resistor 40
2.23 FSR: a) force-to-resistance relation, b) standard interfacing circuit 40

3.1 Example of object data vector 44

4.1 The FG wiring diagram (connection of two other BLDC motors is identical) 46
4.2 KTHand protocol 47
4.3 Voltage levels adjustment schematics 47
4.4 GalilTools programming and real-time monitoring environment 48
4.5 Flowchart of FG Galil control program 51
4.6 Calculations of VG fingers position 52
4.7 Flowchart of VG Galil control program 53
4.8 Format of DSACON32-H communication protocol 54
4.9 Format of data acquisition protocol (respond from controller) . 55
4.10 Data acquisition from 16 sensors example 56

5.1 Functional elements of motion control system 60
5.2 Integrated test platform picture 1 61
5.3 Integrated test platform picture 2 61
5.4 Integrated test platform picture 3 62
5.5 Flowchart of the Integrated test platform. 64

Chapter 1
Introduction

1.1 Problem

Robotic manipulators and gripping devices for power industrial purposes with high level of functionality and dexterity were and still remain in a focus of research in many universities and research laboratories all over the world [3]. One of the interesting tasks is to develop an industrial oriented grasping manipulator which could adjust itself to dynamically changing operational parameters like different shape and size of workpieces. It might be able to perform a given task without human tuning.

Such a gripper device might be able to change its grasping configuration depending on parameters of the object like for example people do. Ability for reconfiguration or in other words flexibility of a gripper could be achieved by using at least three "fingers" [1]. Also different kind of tactile sensors or strain gauge on the finger tips of the gripper fingers could be used together with rotary transducers in the motors to create multipurpose gripping device. Usage of sensors allows to achieve high accuracy in grasping different shapes. Sensitivity to different object parameters, such as weight could be used for automatic adjustment of grasping force. Autonomous adaptation to environmental changes could be achieved by using vision system. The autonomous adaptation means gathering of information about working area (recognition parameters of the object, tracking positions of the arm and the gripper etc.) and dynamic control of movements of the arm and the gripper (position, orientation, open-close-configure moments for the gripper).
There are three separate hardware modules considered in this thesis. They can be described as follows:

1. Flexible gripper (FG)
2. The “ Intelligent arm system” and
3. Vice gripper (VG)

The goal of the thesis is to develop control system for FG and VG and integrate all three modules together. The block diagram of the integrated system layout is shown on figure 1.1.

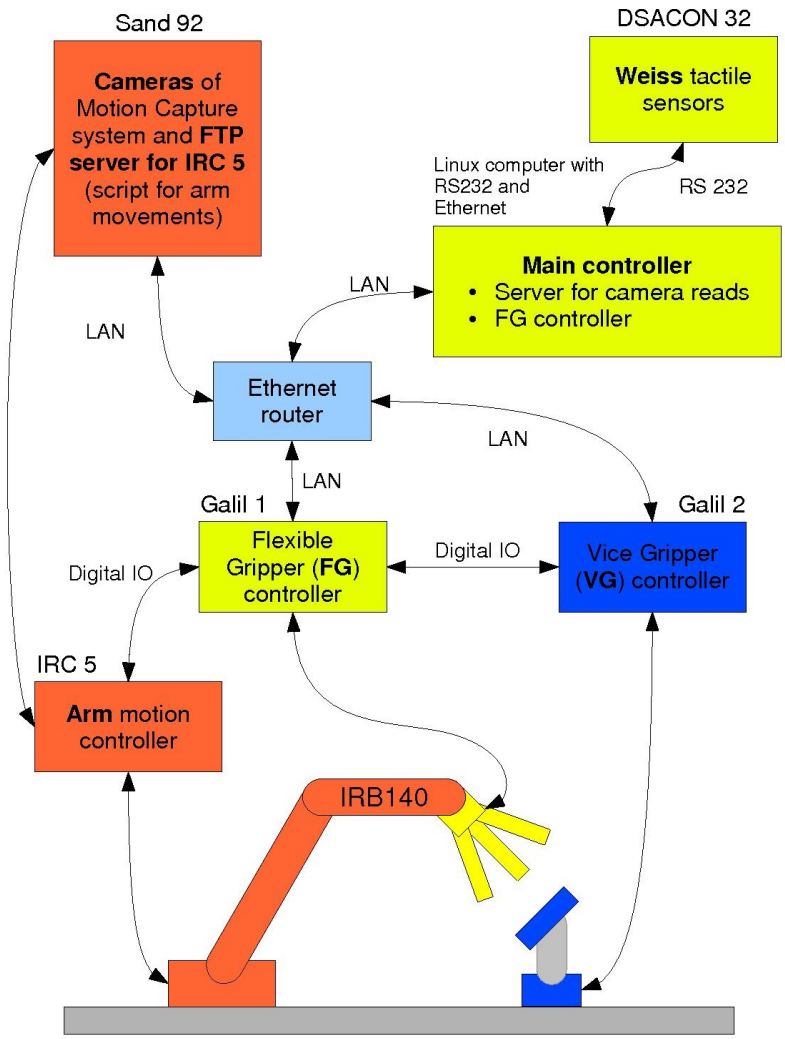

Figure 1.1: The integrated system modules

The main FG control system flow (high-level "Main controller" module on the figure 1.1) reflects integrated system flow (figure 1.2).

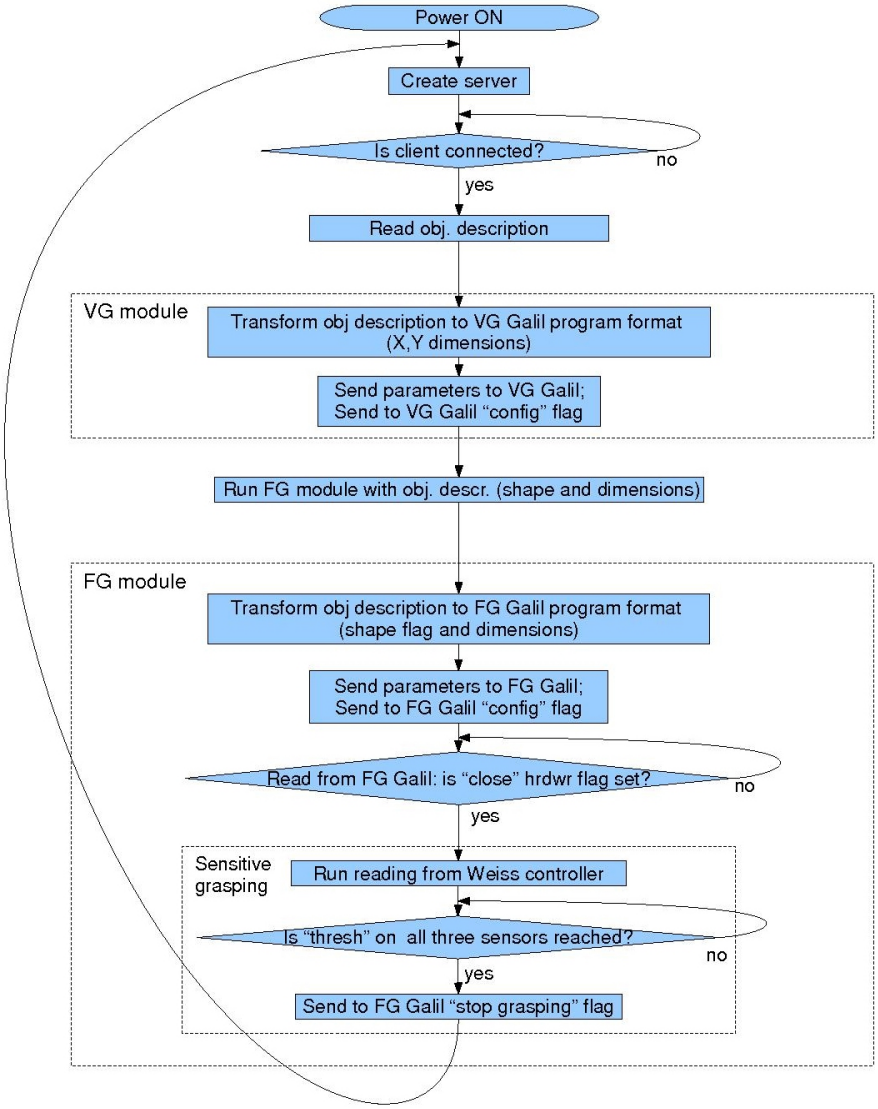

Figure 1.2: The integrated system flow

1.2 Motivation

Let us consider there is a robot manipulator arm at a factory which grasps and moves some workpieces from one point to another. Usually, in that case, arm movements are exactly predefined and simple grippers are used, which allows working only with a few types of details or even just one specific workpiece.

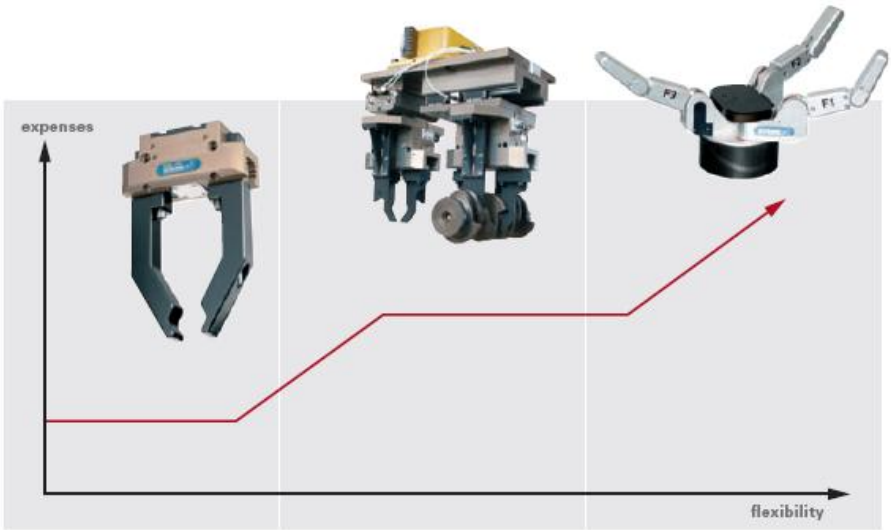

Figure 1.3: Relation of grippers flexibility vs expenses

Different type of changeable gripping devices can be used to increase functionality and effectiveness of manipulator [2]. However, this approach allows working only with limited number of details. At the other hand more universality and flexibility of the gripper requires more expenses for its development. There is a schematic diagram of that dependence on figure 1.3 (source: [4]).

The current project is an attempt to solve a small part of problems with industrial grippers related to their flexibility by developing and implementing a control program for the prototype of costly effective configurable grasping system. Object parameters such as size and orientation can be achieved from video supervision system.

1.3 Objectives

The goals of this thesis are:

- To develop a low-level control system for the new gripper prototype called “Flexible gripper” or FG; the input data for the FG represents a size of the object and its shape.
- To develop a software module for reading tactile sensors data from tactile sensors controller.
- To develop a low-level control system for the second gripper prototype called “Vice gripper” or VG; the input data for the VG also represents a size of the object and its shape.

- To develop a high-level software application; The high level module simply integrates the low level modules with the intelligent arm into one system. It does not have a function in itself.
- To design Integrated test platform in the AASS research lab which illustrates the integrated system and its functionality.

The FG have three fingers with a changeable configuration of two of them, so grasping could be provided at two or three points. The kinematic functionality of all of them are equal. All fingers have sensitive fingertips with tactile sensors.

Tactile sensors are WEISS Robotics [5] DSA 9205 [6] controlled by the WEISS Robotics DSACON 32 tactile sensors controller. Sensitive gripper fingertips could be used to grasp different objects with individual force depending on a database information about the workpieces or perceiving the object image and sensing the weight of the object.

Low-level actuators control is implemented by using a Galil [10] motion controller and the Galil software tools. The communication between high-level control system and the low-level motion control is also implemented by using libraries provided by the Galil.

1.4 Contributions

The contributions can be described as an implementation of the following given subtasks:

1. Development and testing Flexible Gripper control system. It includes a hardware interconnection between mechanical prototype actuators and sensors and Galil DMC 2143 controller and low-level control program to implement grasping movements of the prototype.
2. Development of tactile sensors reading software module by using high-level programming language to communicate with WEISS DSACON 32-H controller. Communication is done via serial RS-232 interface.
3. Integration of Flexible Gripper into the ¨intelligent arm˝ system using KT-Hand protocol.
4. Development of Integrated test platform in AASS lab in purpose of testing and debugging a control program and presentation of the results of the project. During experiments on Integrated test platform all necessary hardware work (development of intercommunication electronic board with voltage level matcher) was done as a part of FG integration.

5. Development of the control program for the second gripper prototype, named Vice gripper or VG. VG was added into the Integrated test platform. The control program realizes an integration with the ¨intelligent arm˝system.

The overall goal of the project was to develop a complete control system for the Flexible Gripper prototype. The complete control system should have features such as sensitive grasping based on tactile sensors, communication with the ¨intelligent arm˝system and reading of the object information from the motion capture system. The total implementation of the control system exists only as a number of modules.

The future work is to develop a high-level software application. It should implement features such as:

1. reading of an object description from the motion capture system (done partially)
2. translation of the object description into a specific format for FG and VG control systems
3. uploading the object description into FG and VG controllers
4. combining all these new features with the already existing tactile sensors module to achieve the ¨precise sensitive grasping˝.

1.5 Thesis structure

This thesis is divided into 6 chapters.

Chapter 1 discusses briefly the state of the art in the gripper devices research and development. It also formulates the purpose of the thesis, task requirements and contributions of the author.

Chapter 2 describes a hardware equipment and tools used to implement Flexible gripper functionality required by thesis task. Some technical aspects of the ¨intelligent arm˝system are also mentioned as it is a part of gripper integration task of the thesis.

Chapter 3 describes a method for object exploration by a computer vision system implemented in AASS lab. Also in this chapter there are attempts to familiarize with a robot autonomous grasp planning methods based on a demonstration of human grasping techniques as a similar technique may be used to input control signals of the flexible gripper device.

Chapter 4 explains how all parts of the thesis, hardware and software, were done and what kind of methods and techniques were used to accomplish thesis tasks.

Chapter 5 describes an implementation of the thesis experiment as a part of the ˋintelligent arm˝system. Shows a video evidence of a real work of the developed system. Guides through experimental platform process and some of its technical aspects.

Chapter 6 summarizes the work done and main problems appeared during thesis implementation. It also designate related future work.

Chapter 2
Mechanical gripper prototype and control hardware tools

In this chapter all of the hardware tools and their main properties used in this thesis are described. Mechanical gripper prototype and hardware tools were given by the supervisor and this thesis's aim was to develop software control modules for existing hardware. Some of the kinematical aspects and functional capability of the gripper are presented in this chapter. Galil motion controller was given as a control device to implement a control program and therefore the description of some of the main parts of hardware and computational electronics functionality are also necessary in this thesis. Specific tactile sensors and their controller description are also given in this section. One of the tasks of the thesis is to integrate a mechanical gripper into the ¨intelligent arm˝system, hence some parts of this system are also mentioned.

All hardware tools and devices are given as a starting point for the development software control system and description in this section is only related to the given existing hardware and tools. All contributions, mentioned in Chapter 1 are described in Chapter 4 of this paper.

2.1 Flexible gripper mechanics

A New prototype of a multi functional adjustable gripper (FG) was developed in AASS research lab. The gripper has three fingers with a changeable configuration of two of them and fixed relative to FG frame position of third finger. Schematic diagrams of different configurations of the FG are shown on figure 2.1

On figure 2.1, a detail with an ¨upside down T˝shape with a left hatching is a FG frame. All three fingers move in fingers' bases. Fixed finger base is on the top of the FG frame and two adjustable fingers' bases can be rotated in any

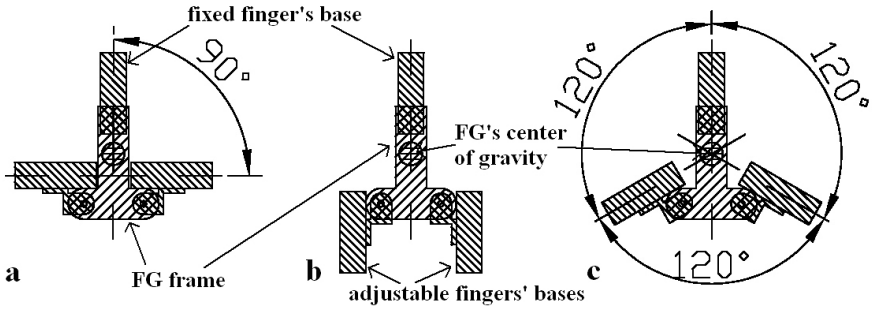

Figure 2.1: FG configuration (Top view)

angle position between 90 and 180 degrees with respect to FG frame. Photos of FG prototype are shown on figure 2.2.

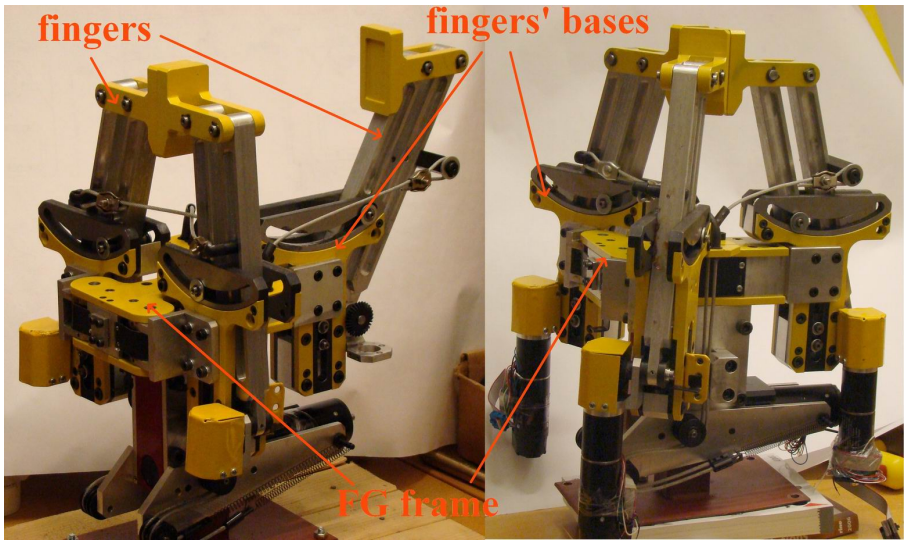

Figure 2.2: FG prototype

All three fingertips move in a direction perpendicular to FG frame plane. This state-of-the-art design allows to grasp any type of objects and at the same time there are no additional joints, which surely might complicate the construction and increase overall cost of development.

FG has only one motor on each finger to move them inside its bases and one motor to rotate adjustable fingers bases relative to FG frame. Since there is only one configuration motor, adjustable fingers are depend on each other and can

be moved only together. Therefore, as also shown on figure 2.1, there are three main possible configurations of the FG fingers, which are the most interesting:

1. to grasp relatively small rectangular-shape objects (< 200 mm at the longest side, because this is a distance between two fingers in configuration b), figure 2.1) at two points,
2. to grasp big rectangular-shape objects (up to about 400 mm at the shortest side - twice of maximum range of one finger) at three points
3. to grasp round-shape objects from 30 mm to 400 mm in diameter also at three points.

Schematic diagram of three main configurations can be seen on figure 2.3.

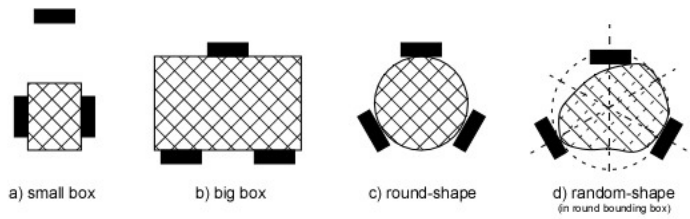

Figure 2.3: Three grasp points configurations

DC motors from Maxon Motor GmbH have been used as actuators of fingers and configuration. Three brushless motors (BLDC) are used for fingers and one brushed motor is used for fingers configuration. Motors selection has been done by the constructor of the prototype.

There are no mechanical contacts between the voltage source and the motor's rotating components in brushless motor. This fact means reducing of the potential for failure. It also reduces the electromagnetic and radio frequency interference (EMI and RFI). Schematic diagram of brushless DC motor is shown on figure 2.4.

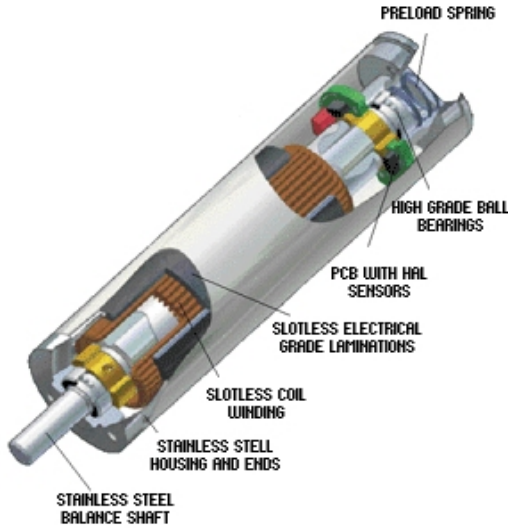

Figure 2.4: Brushless DC motor components

The commutation function is performed by Hall effect magnetic sensors. A Hall effect sensor is a semiconductor device where the electron flow is affected by a magnetic field perpendicular to the direction of current flow. It looks like a four terminal variable resistor network. The voltages at the two outputs are complementary. Application of a magnetic field to the sensor causes a small voltage change at the output. This 3-lead device may directly drive the power transistor feeding a phase winding. The sensor must be mounted close to the permanent magnet rotor to sense its position (see figure 2.5).

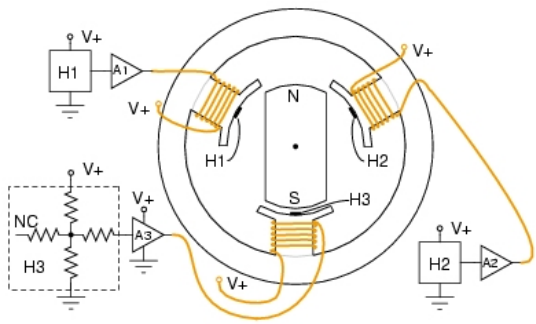

Figure 2.5: Three phase brushless DC motor mechanism

In the used Maxon BLDC the Hall outputs connected to the Galil controller which then generates three power phases to feed three motor's windings. The

schematic diagram of the wiring between Maxon BLDC and Galil controller is shown on figure 2.6.

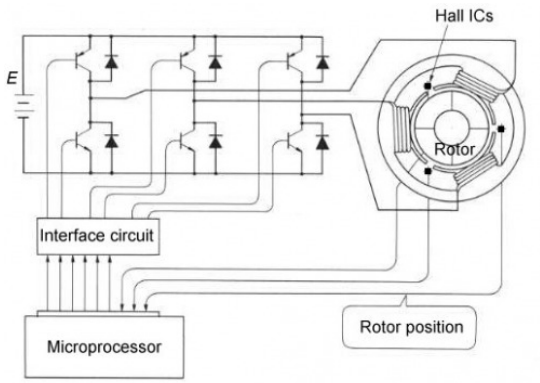

Figure 2.6: The schematic diagram of the connection of three phase brushless DC motor to the controller

All motors equipped with high precision optical encoders with resolution of 500 counts per motor spindle revolution (CPR). All motors are also equipped with decreasing gearboxes with 392:1 ratio and therefore the total resolution of actuator is about 190000 counts per revolution. Practically it gives a linear resolution about 138 counts of encoder per mm of finger movement (data acquired during experiment on assembled gripper system). However, mechanical construction of all fingers has too big backslash between motor rotation and fingers movements. This fact allows to use motor encoders as position estimators of a finger. However it gives only a rough estimation. Configuration actuator is also equipped with a gearbox. The total resolution of configuration becomes 218 counts per degree (experimental data), which in spite of a mechanical backslash is very precise.

Each finger is equipped with a limit switch or ˝switch˝ in a maximum ˝open˝ position. The end switch for configuration axis is located in ˝180˝ degrees position (see configuration b), figure 2.1). They are used to determine a reference point and also protect a controller from the overload when maximum position is reached. Two different types of the limit switches are used. The end switches for fingers are inductive proximity sensors. The end switch for configuration is a precise mechanical switch (circuit breaker).

Inductive proximity sensor detects the presence of metallic objects. It has four main components:

1. a coil
2. an oscillator

3. a detection circuit
4. an output circuit

The oscillator generates a fluctuating magnetic field in the shape of a doughnut around the winding of the coil that is located in the device's sensing face (see figure 2.7).

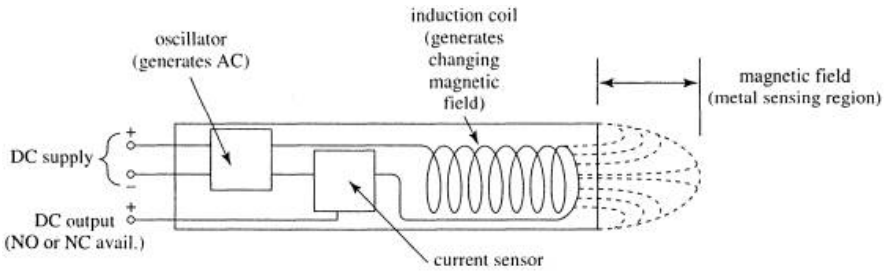

Figure 2.7: Schematic diagram of Inductive Proximity sensor

The presence of metal in the operating area causes a reduction of the Inductive sensor's own oscillation field. The sensor's detection circuit monitors the oscillator's strength and triggers an output from the output circuitry when the oscillator becomes reduced to a sufficient level. The operating distance of the sensor depends on the coil's size as well as the target's shape, size and material. The example of the inductive proximity sensor is shown on figure 2.8

Figure 2.8: Inductive Proximity sensor

Mechanical limit switch is a device that uses physical contact to detect the target. A typical limit switch consists of a switch body and an operating head. The switch body contains electrical contacts to energize or de-energize a circuit. The operating head incorporates a lever arm or plunger. This is also called an actuator. The example of the mechanical limit switch is shown on figure 2.9

Figure 2.9: Mechanical limit switch

Interfacing of the limit switches of two types of sensors to the Galil controller is different. The inductive sensor is an active element and needs +24V power supply. Its output signal is also +24V. This fact requires a voltage level adjustment, because the Galil controller uses +5V signals. The schematic diagram of +24V to +5V voltage level aligner is shown on figure 2.10.

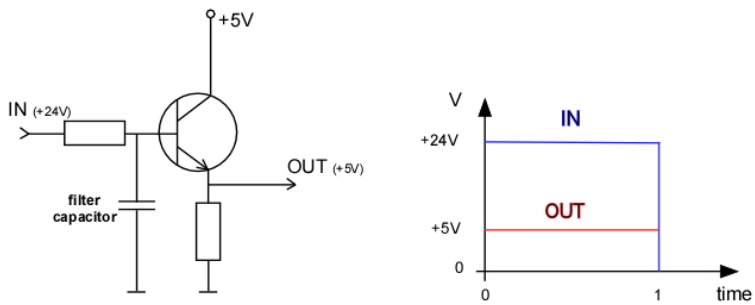

Figure 2.10: Voltage level adjustment circuit

The mechanical switch is just a physical contact. The standard Galil +5V source is used to interface it to the controller. Therefore interfacing consists of a connection of its wires between of the +5V source and one of the digital inputs of the controller.

2.2 Galil motion control system

The motion control system consists of a motion controller, an amplifier, motors, hall sensors, encoders as well as the end switches, see figure 2.11.

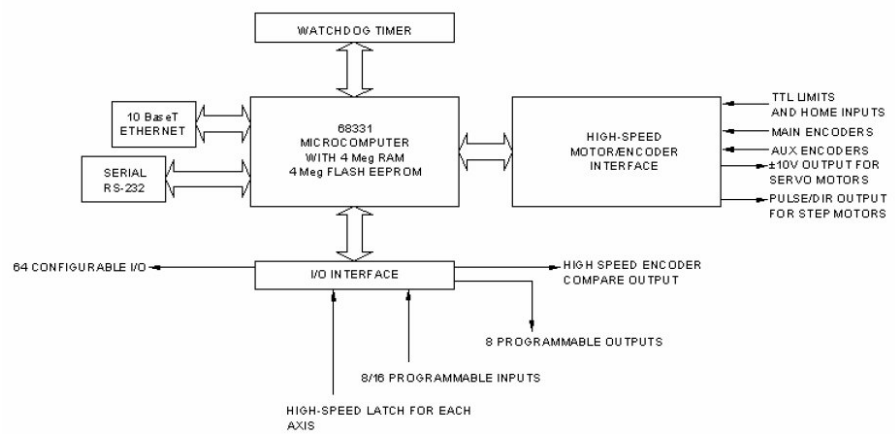

Figure 2.12: DMC-2143 functional elements

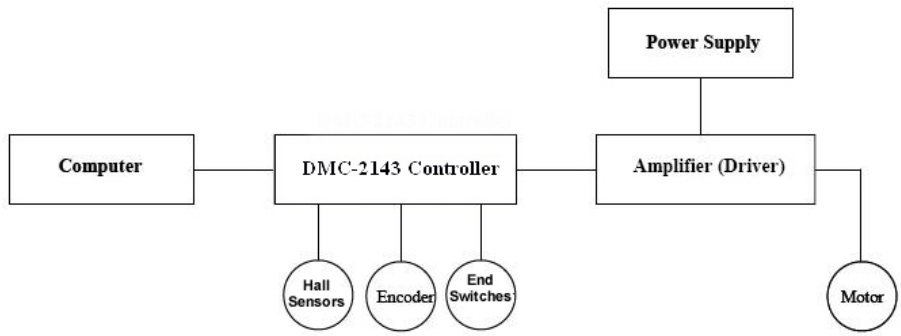

Figure 2.11: Elements of motion control system

All following concise description parts and illustrations of elements of the motion control system have been taken from DMC-2143 controller user manual [13] and DMC-2143 accessories manual [11].

2.2.1 DMC-2143 controller

DMC-2143 controller includes four main sections: 1) main processing unit, 2) motor interface, 3) general I/O interface and 4) high-level communication interface. The schematic diagram of DMC-2143 controller is shown on figure 2.12.

The main processing unit is a specialized 32-Bit Motorola 68331 Series Microcomputer with 4 Meg RAM and 4 Meg Flash EEPROM. The RAM provides a memory for variables, array elements and application programs. The flash EEPROM provides a non-volatile storage of variables, programs, and arrays.

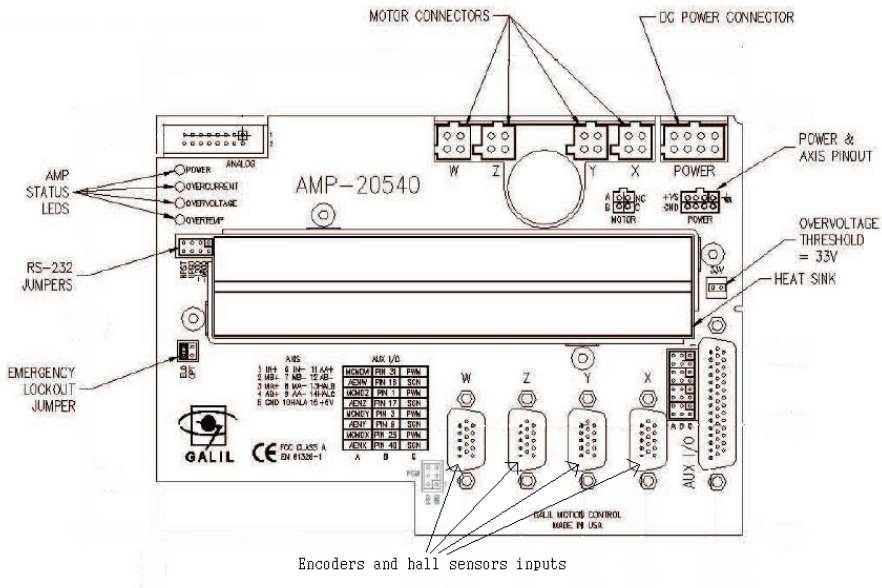

Figure 2.13: AMP-20540 amplifier

It also contains the DMC-21x2/21x3 firmware.

In Motor Interface module Galil's GL-1800 custom, sub-micron gate array performs a quadrature decoding of each encoder at up to 12 MHz. For standard servo operation, the controller generates a +/-10 Volt analog signal (16 Bit DAC).

The communication interface consists of a RS-232 port (up to 19.2Kbaud) and 10 BaseT Ethernet port. The DMC-2143 provides an interface circuitry for 8 TTL inputs, 8 TTL outputs. Unused auxiliary encoder inputs may also be used as additional inputs (2 inputs / each axis). The general inputs may also be used as high speed latches for each axis. A high speed encoder compare output is also provided. The DMC-2152 through DMC-2182 controller provides an additional 8 TTL inputs and 8 TTL outputs.

2.2.2 AMP-20540 amplifier

For each axis, the power amplifier converts a +/-10 Volt signal from the controller into current to drive the motor. The layout of AMP-20540 amplifier is shown on figure 2.13.

The four-axis AMP-20540 capable of handling 500 watts (7 Ampers) of continuous power per axis and 10 Ampers in peak. The amplifier is a brush/brushless trans-conductance PWM amplifier. The AMP- 20540 Brushless drive modules are connected to a DMC-2143 controller via the 96 pin DIN connector. The standard amplifier accepts DC supply voltages from 18-60 VDC. The AMP-20540 provides for the addition of 8 analog input to the DMC-2143. The analog inputs accept +/- 10 V input and have a resolution of 12 bits. Nominal Amplifier Gain 0.4, 0.7, and 1.0 A/V, switching Frequency 60 kHz, Brushless Motor Commutation angle 120 degrees.

2.2.3 Theory of operation

A typical motion control system consists of the elements shown in figure 2.14.

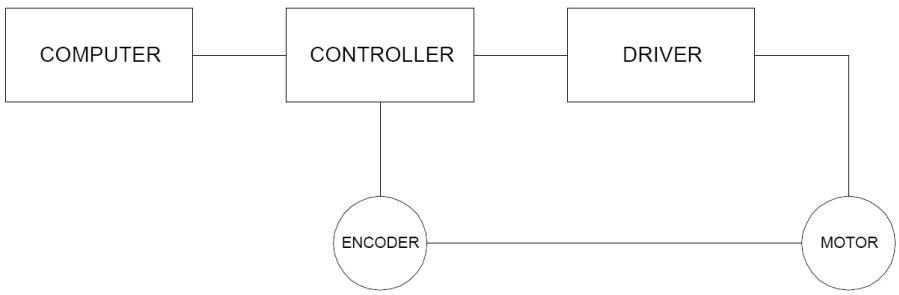

Figure 2.14: Elements of Servo System

The operation of such a system can be divided into three levels (source: [14]):

1. Closing the Loop
2. Motion Profiling
3. Motion Programming

The first level, the closing of the loop, assures that the motor follows the commanded position. This is done by closing the position loop using a sensor. The motion profiling is the generation of the desired position function. This function, R(t), describes where the motor should be at every sampling period. Profiling and the closing of the loop are independent functions. The profiling function determines where the motor should be and the closing of the loop forces the motor to follow the commanded position The highest level of control is the motion program. This can be stored in the host computer or in the controller. This program describes the tasks in terms of the motors that need to be controlled, the distances and the speed. Functional element of motion control system are shown on figure 2.15.

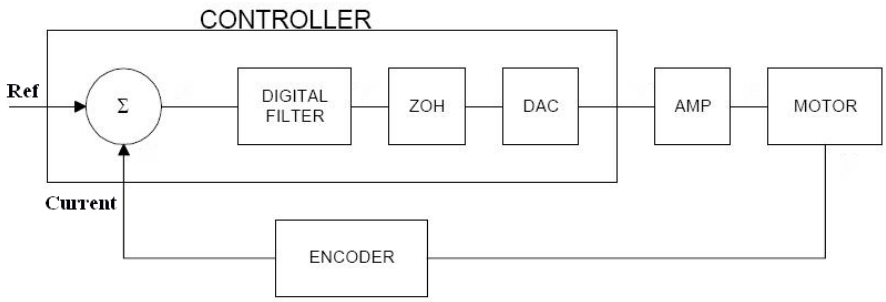

Figure 2.15: Functional elements of motion control system

1. Amplifier

   The motor amplifier may be configured in three modes:

   Voltage Drive The amplifier is a voltage source with a gain of K_v [V/V].

   Current Drive The current drive generates a current I, which is proportional to the input voltage, V, with a gain of K_a.

   Velocity Loop The motor driver system may include a velocity loop where the motor velocity is sensed by a tachometer and is fed back to the amplifier.

2. Encoder

   The encoder generates N pulses per revolution. It outputs two signals, Channel A and B, which are in quadrature. Due to the quadrature relationship between the encoder channels, the position resolution is increased to 4N quadrature counts/rev. The model of the encoder can be represented by a gain of

   $$K_f = \frac{4N}{2\pi}, [count/rad]$$

   For example, a 1000 lines per revolution encoder is modelled as

   $$K_f = 638$$

3. DAC

   The DAC or D-to-A converter converts a 16-bit number to an analog voltage. The input range of the numbers is 65536 and the output voltage range is +/-10V or 20V. Therefore, the effective gain of the DAC is

   $$K = \frac{20}{65536} = 0.0003, [V/count]$$

4. Digital filter

   The digital filter has three element in series: PID, low-pass and a notch filter. The transfer function of the filter. The transfer function of the filter elements are:

$$PID : D(z) = \frac{K(Z-A)}{Z} + \frac{CZ}{Z-1}$$

$$Low-pass : L(z) = \frac{1-B}{Z-B}$$

$$Notch : N(z) = \frac{(Z-z)(Z-\overline{z})}{Z-p)(Z-\overline{p})}$$

   The filter parameters, K, A, C and B are selected by the instructions KP, KD, KI and PL, respectively. The relationship between the filter coefficients and the instructions are:

$$K = (KP + KD) * 4$$

$$A = \frac{KD}{(KP + KD)}$$

$$C = \frac{KI}{2}$$

$$B = PL$$

   The PID and low-pass elements are equivalent to the continuous transfer function G(s).

$$G(s) = (P + sD + \frac{I}{s}) * (\frac{a}{S+a})$$

$$P = 4KP$$

$$D = 4T * KD$$

$$I = \frac{KI}{2T}$$

$$a = \frac{1}{T} ln(\frac{1}{B})$$

   where T is the sampling period.

5. ZOH

   The ZOH, or zero-order-hold, represents the effect of the sampling process, where the motor command is updated once per sampling period. The effect of the ZOH can be modeled by the transfer function:

$$H(s) = \frac{1}{1 + \frac{sT}{2}}$$

   If the sampling period is T = 0.001, for example, H(s) becomes:

$$H(s) = \frac{2000}{(s + 2000)}$$

   However, in most applications, H(s) may be approximated by a unit gain.

2.3 Tactile sensors system

Description of WEISS DSA 9205 tactile transducer has been taken from a user manual [6] of the sensor. Description of WEISS DSACON 32-H controller has been taken from a user manual [8] of the controller. Tactile sensor system consists of three DSA 9205 sensors arrays (6 by 14 sensors) and DSACON32 sensor controller. Schematic diagram is shown on figure 2.16.

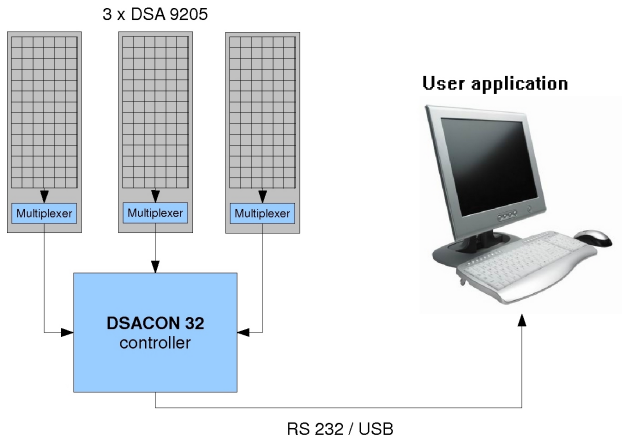

Figure 2.16: Tactile sensors system

The sensor controller represents the link between a tactile transducer of Weiss Robotics and a high-level user application. It digitalizes the measured data of the connected transducer and then it can be read on demand via standard computer communication interface. This controller allows to use one of two serial interfaces RS232 or USB. It consists of a microcontoller to read the data from tactile matrices and to handle communication interface. It uses specific communication protocol and therefore requires a specific user application to send commands into controller and translate acquired serial data into values of pressure. In this project RS232 interface were used. Controller has fixed parameters of RS232 data transferring interface which cannot be changed. The actual parameters are: speed at 115,2kbit/s and 8 bits data packets. However, a digital resolution of the analog-to-digital converter of controller is 12 bit and each tactile sensor reading represented as two bytes in a communication protocol. More detailed description of the communication protocol and data acquisition discussed in Chapter 4 of the thesis. Controller requires stabilized single +24V power supply and designed specially for industrial applications. More technical parameters of DSACON32-H controller are represented in the table on figure 2.17:

Supply voltage	10 to 28 VDC, typ. 24 V
Power requirements	0,6 W at 24 V
Integrated interfaces:	RS232 (115,2 kBit/s), USB 2.0 (full), CAN-Bus
Maximum transducer size	2x 1.024 sensing points
Maximum sampling rate	1,2 MSPS
Resolution	12 Bits
Signal to Noise Ratio	68 dB (at f = 20 kHz)
Power supply for the transducer	2x 5 V / 150 mA
Dimensions (L x W x H)	95 x 85 x 30 mm
Weight	180 g
Operating temperature range	0 to 40 °C
Sensor port connector	2x 14 pos. MDR-Connector

Figure 2.17: Table of DSACON32-H parameters

Compact design and standardization of power supply and communication interface make it suitable for wide variety of applications. Image of DSACON 32-H is shown on figure 2.18.

Figure 2.18: DSACON 32-H controller

The tactile transducer DSA 9205 is shown on figure 2.19. It was specially developed for use in robot grippers. It consists of a 6 x 14 point pressure sensitive matrix with a spatial resolution of 3.4 mm. Due to its compact dimensions of 24.4 x 51.4 x 5.4 mm (L x W x H), the DSA 9205 is suitable for many applications. The integrated signal conditioning leads to a high noise immunity even in rough industrial environments. The DSA 9205 is electrically contacted by a connector on the rear of the module. This simplifies the exchange of the module by user.

Figure 2.19: DSA 9205 tactile transducers

Specifications of DSA 9205

- Power supply: 5 V, 10 mA
- Number of sensing points: 6 x 14
- Spatial resolution: 3.4 mm
- Measurement range: 250 kPa
- Sampling rate: 230 fps
- Temperature range: 0 to 40 C
- Outer materials: Stainless steel, silicone rubber
- Protection class: IP65 (mounted and sealed with o-ring)
- Dimensions (see drawing): 24.4 x 51.4 x 5.4 mm

2.4 The ¨intelligent arm˝system

The ¨intelligent arm˝system consists of IRC5 ABB arm controller, IRB 140 ABB robotic (see figure) arm and motion capture system, which has five cameras located above the arm working area and control unit (see figure). Motion capture system allows to recognize objects and arm movements within working area of the arm with very high resolution (about 2 millimeters).

IRB 140 ABB arm is capable to safely carry a payload up to 5 kg. IRC5 ABB arm controller has digital I/O interface to react on external events and control external devices. Interface requires industrial standard +24V voltage levels.

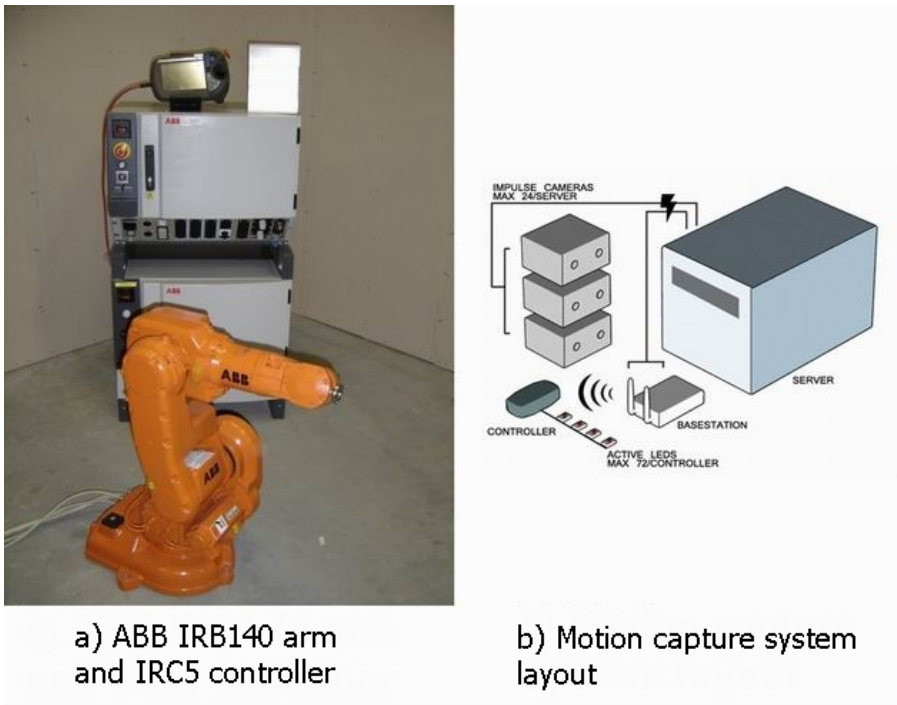

Figure 2.20: Components of the "Intelligent arm system"

An integration of the gripper prototype with the ¨intelligent arm´system requires only intercommunication between IRC5 ABB arm controller and Galil motion controller. As were mentioned above, Galil controller also has a number of digital I/O channels to communicate with external devices and to control a program flow. However, Galil motion controller has +5V digital interface, while IRC5 +24V. Therefore, from technical point of view, the only problem is to convert voltage levels as required for each controller. Integration process details, both from programming and electrical points of view, are described in Chapter 1, since current chapter covers only given hardware tools and some of their necessary parameters.

2.5 Vice gripper

Vice gripper or VG is also another prototype of a gripper developed in AASS lab. It also operates under Galil motion controller. VG is used in the Integrated test platform and should work in pair with FG. Schematic diagram of VG movements abilities is shown on figure 2.21.

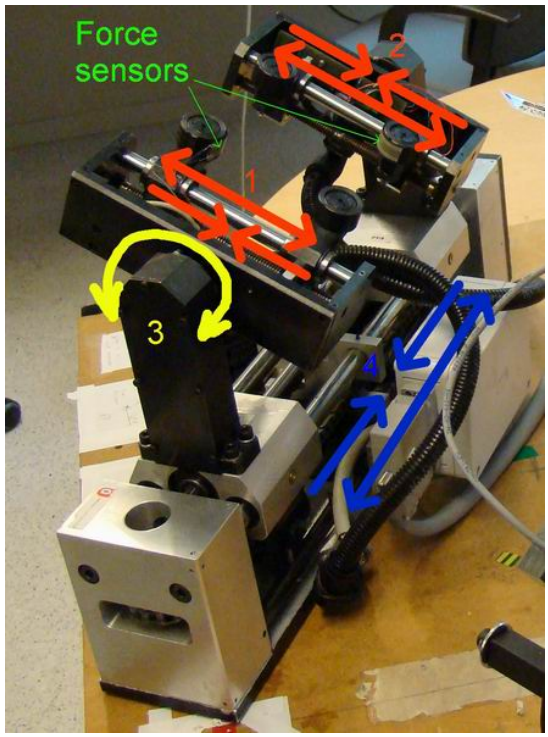

Figure 2.21: Vice gripper degrees of freedom

There are four independent motions possible. VG can open and close fingers separately from each other (motions 1 and 2 on figure 2.21, named vertical movements) and therefore provides a possibility to grasp objects in four or three points (when both fingers on one side are brought together). Horizontal movement is shown on the figure in blue color. Capture of objects is realized by simultaneous movements in both, vertical and horizontal planes. Planes of motions 1 and 2 are connected to each other constructively. Motion number 3 (on figure in yellow color) reflects a rotation of those planes. Brushless motors with optical encoders are used for all movements. Two fingers (green pointers on figure) are equipped with a force-sensing resistors (FSR and "Force Sensing Resistor" are trademarks of Interlink Electronics, Inc.).

Force Sensing Resistor (FSR) is a polymer thick film (PTF) device which exhibits a decrease in resistance with an increase in the force applied to the active surface. Its force sensitivity is optimized for use in human touch control of electronic devices. FSRs are not a load cell or strain gauge, though they have similar properties. FSRs are not suitable for precision measurements. [7]. The image of the FSR is shown on figure 2.22 (from [7]).

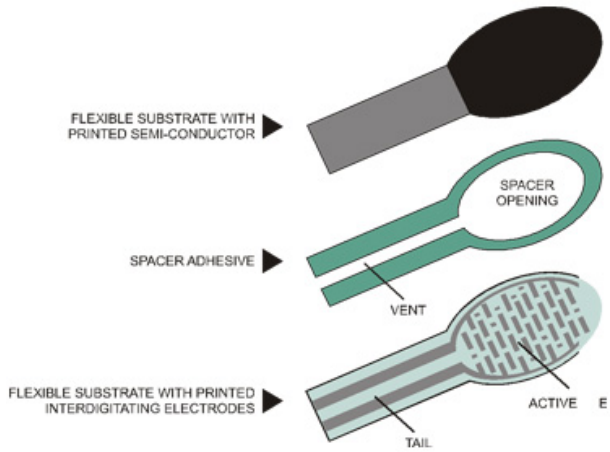

Figure 2.22: Force-sensing resistor

FSR force to resistance relation and standard interfacing circuit are shown on figure 2.23 (from [7]).

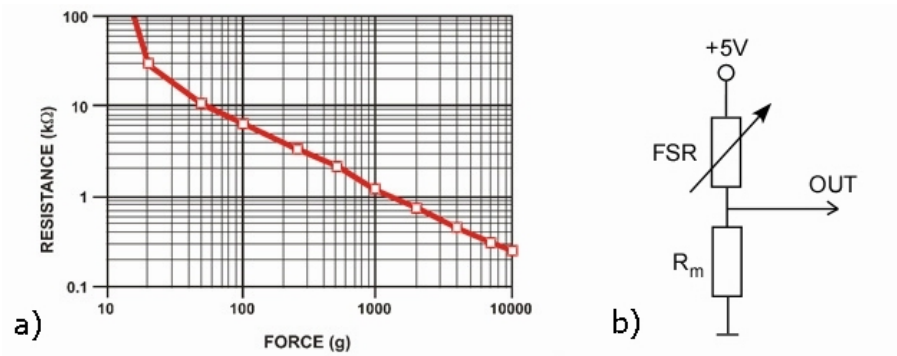

Figure 2.23: FSR: a) force-to-resistance relation, b) standard interfacing circuit

The two FSRs are used in VG as touch sensors and connected to the Galil controller via two separate analog inputs. The Galil DMC-2143 controller has an Analog-to-Digital converter on every analog inputs and values can be read in the program. The control program of VG reads the inputs periodically and stops the grasping when values on both sensors exceeds threshold. The value of a threshold can be changed manually during the program flow to adjust the grasping force.

Chapter 3
Vision guidance system for grasping by demonstration

Programming by demonstration (PbD) involves different methods where the user teaches the robot tasks by demonstrating them. It remains a key research topic in robotics [33] mainly because it greatly simplifies the programming process [38].

3.1 Automatic grasping based on demonstration

For training systems that deal with object grasping, both visual and tactile feedback is required (statement in this section is based on a related work described in [35]).
Following features are commonly used for automatic grasping:

1. position, orientation, and shape of the object is known [21].
2. boundary contour of the object should be possible to extract; counter is necessary for a planar grasp [19].

The work on a contact-level grasps synthesis focusses mainly on finding a number of contact points in spite of the gripper design [16].
The method mentioned in [21] represents objects as sets of shape primitives such as spheres (cylinders, cones) and boxes. Each primitive is associated with several rules to generate a set of hand configurations and preshapes.
The choice of the suitable grasp is learned from the human. It is based on defining the set of human hand configurations with respect to the specific object. However, grasp preshapes are generated based on a recognition of human grasps. Therefore they are also related to the object pose and shape and not only its outer contour. The FG controller provides support for this methodology. The data coming from human demonstrations is used to command the FG. This data carries an information about an object size and shape. The pose

of the object is in other words its orientation with respect to the gripper. It is obviously a part of the arm motion control.
One of the objective of the thesis was to accommodate the motion capture system as a source of information which supports the grasping process. The control module of the FG is designed to support visual guidance. It can utilize the object information coming from a vision system.

3.2 Objects exploration and features extraction

The system of object description was implemented in AASS research lab. Further description of the system is adapted from source [24].
A lot of work has been done in the area of visual object recognition and modeling for grasping. For example, [25] presents a learning algorithm for grasps prediction for parallel gripper based on 2D images. Other authors [26], [27] try to estimate most stable grasping points for multi-fingered hands based on different vision systems and grasp stability criteria. In [28], [29] a method for pre-grasp selection is proposed by decomposing object shape into minimum bounding boxes.
However, the robustness of vision based methods to object recognition and localization is compromised if visual clues are absent or in cluttered environment. Some of the mentioned vision problems are solvable by using an active exploration. And a human demonstration based approach has clear advantages.

3.2.1 Motion capture System

A variety of human tracking devices have been used, following [30], the most popular one due to its robust and accurate hand pose acquisition is a data glove. In [31] the posture of a bare human hand is tracked for grasp acquisition using a camera. Mixing together some vision based methods, should provide very robust grasping oriented recognition system.
Proper experimental setup is needed to register robustly and efficiently a human demonstration . The system should fulfill two main requirements, firstly, acquire of the configurations of two hands in real-time, secondly, it should allow the user to explore an object in natural and convenient way. For this reason in our setup a vision based PhaseSpace motion capture system [32] for human hands tracking hasbeen used.
The advantages of the system are high accuracy and sampling rate, robustness to changing light conditions and, in contrast to 2D- and 3D vision methods, it allows exploration of objects without assumptions of any specific visual clues. Of course, it suffers from occlusion as any other vision based system.
To let the user explore an object fast and freely two gloves are used. Each glove has nine diodes and five force sensing resistors used as tactile sensors emplace on the glove. Each diode blinks in a unique pattern that can be identify and its 3D position is tracked using a set of five stereo cameras placed around the working area.

3.2.2 Grasp-related features extraction

An Operator picks an object of interest from the table and handles it freely using both hands. The collected point cloud appears on the screen so a visual feedback is possible. Resultant data is then clustered and grasp-related features are associated with the object.

Object exploration The whole exploration consists of a sequence of fingertip grasps registered by the tactile sensors and the motion capture system. During the manipulation atom sets of points, that each represents separate, fingertip grasp, are collected. Because the user is allowed to change the holding hand, a transformation between grasps is calculated to keep all points in a one object coordinate frame.

Features extraction Resulting data from the object exploration step is a collection of atom point sets that together create 3D point cloud. The point cloud is sparse, so only rough deduction about the object geometry is possible, however it is good enough to show graspable regions on the object surface. Moreover, a human way of approaching the object is captured in collected data. Atom sets are firstly clustered into separate graspable regions - bodies. Secondly, for every grasp an approach vector and associated with it grasp oriented bounding box (GOBB), that bounds all points that belong to the same body, are generated. As a result, the object is approximated with a set of overlapping GOBB with respective approach vectors. Bounding box has been chosen as an approximation primitive, however, such fitting can also be done with other primitives like cylinders or spheres.

3.3 Flexible gripper control program

In this thesis FG controller receives object shape description from the Motion Capture System. Only two different shape primitives of an objects are enough in the most cases to describe all variety of the real objects' shapes and perform stable grasping. That means that only three main configurations of the gripper are required (see example on figure 2.3 in Chapter 2).

Control program provides control procedures for Flexible Gripper actuators (motors). It implements autonomous movements of gripper fingers such as:

- self-configuration: sequence of full opening, closure and rotation of the fingers in order to find out numerical representation of possible areas of motions
- recalculation of input data about the size and shape of the object into positions of each finger and rotation angle of two adjustable fingers

- implementation of a hard stop: error of the motor's position is used to judge about the finger motion (if the object hampers a motion of the finger, an error will increase considerably); control program checks error values periodically and stops the motor if its error exceeds threshold; the object is considered successfully grasped if all three motors are stopped

Control signals for Flexible gripper system can be retrieved from the information provided by the Motion Capture System. As were mentioned in the previous section it uses object shape boundaries to describe parameters of the object. In most cases objects present in two main shapes: a box (cube, parallelepiped) and a cylinder (sphere, cone). For the given gripper functionality it means only three main configurations:

1. a small box or parallelepiped with the biggest side up to 200 mm
2. a cylinder with a maximum diameter of 300 mm
3. a big box or parallelepiped with a maximum dimension of the shortest side 300 mm

Orientation of the object with respect to the gripper is a part of the "Intelligent Arm system" or provided by the motion capturing system and assumed always correct. Correct means that the main plane of the gripper is oriented in parallel to the sides of the rectangular objects. The dimensions of objects should be given in a specific format. For example the first value represents a dimension of the main side (or plane) of the object, the second value represents a dimension of the other side. The schematic diagram of such example is shown on figure 3.1.

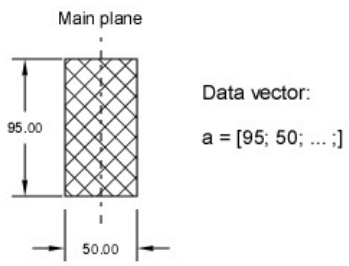

Figure 3.1: Example of object data vector

Chapter 4
Implementation

4.1 Hardware

Since FG was provided fully assembled, no mechanical work was necessary. However, connection between FG's actuators and Galil motion control system needed to be done and was a part of this thesis. Due to the gripper complexity and the spatial constraints, the cabling design and realization appeared to be a rather challenging (and time-consuming) task.

4.1.1 Flexible gripper wiring

The nearest place where quite big and heavy controller could be placed is the shoulder of the arm. Therefore numerous complicated wiring were done.

Every brushless motor requires three power lines, five lines for hall sensors (three signal lines, +5V low power line and a common "ground" line) and four lines for encoders (two signal lines, +5V low power line and ground). Since hall sensors and encoders use the same power voltage and do not require high power to operate, these four separate pairs can be combined into only one pair to reduce a total amount of cables. Configuration motor is brushed and requires only a pair of power lines. Since encoder for this axis is the same, it also requires one pair of signal lines. Total number of lines needed to connect fingers' motors are 30 (3x3 brushless motors phases + 2 brushed motor lines + 3x3 hall sensors + 4x2 encoders + 2 low power +5V line). In addition to described wiring lines there are four end switches lines required (four separate signal lines and one pair of +24V low power line). Therefore total number of separate lines is 36.

All motors, hall sensors, encoders and end switches cables were connected to the Galil controller through the interconnection board. A small board was necessary to reduce total number of separate cables and to increase reliability, convenience of use and maintainability of wiring. Schematic diagram of FG wiring

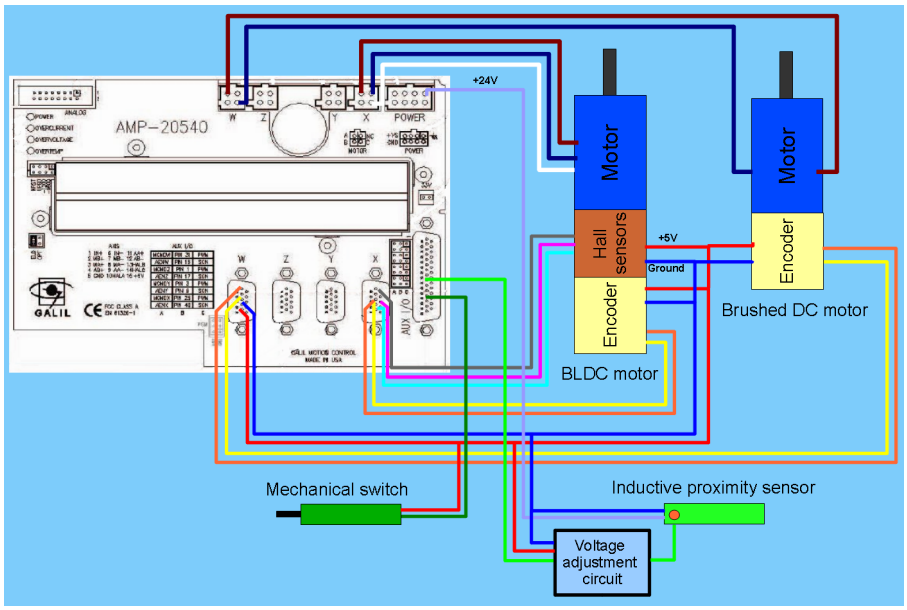

Figure 4.1: The FG wiring diagram (connection of two other BLDC motors is identical)

is shown on figure 4.1.

There was some problems during testing and debugging of the wiring. Few short circuits and missrouting were eliminated. Also a big issue was an interfering of the signal lines with the magnetic switches lines. The noise was produced by +24V signal lines running in parallel with the other lines. After a few attempts to figure out the reason of the noise, analog filters were added to resolve this problem. Therefore several phases of corrections during a week or so were done before everything began to work properly.
Connection of tactile sensors system needed only one RS232 communication cable to connect DSACON 32-H controller to computer and a separate power line.

4.1.2 Flexible gripper hardware integration

Communication between FG and the ¨intelligent arm˝controller IRC5 were done by three-line digital interface. This interface provide possibility to use KTHand protocol (described in [36]) to control FG with IRC5. Schematic description of KTHand protocol is shown on figure 4.2.

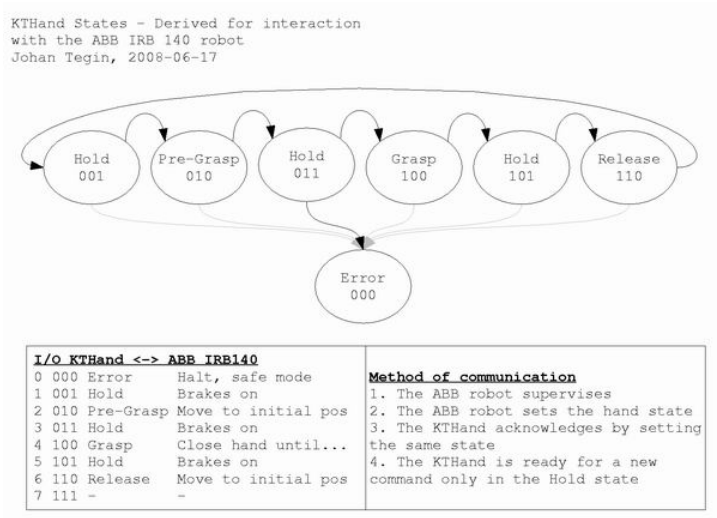

Figure 4.2: KTHand protocol

However IRC5 controller uses +24V level of the I/O signal while Galil controller uses +5V signals. Voltage levels adjustment board was developed. It also became a problem, because low-voltage (+5V) signals had to be transformed into a high-voltage (+24V) and vice versa at the same time. It required a complicated routing and two separate power supplies. It took another day or so until it began to work. Schematic diagram of the voltage adjustment elements are shown on figure 4.3.

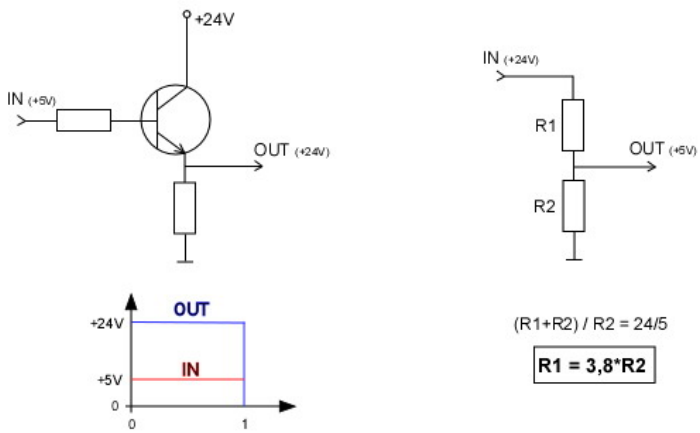

Figure 4.3: Voltage levels adjustment schematics

Other connection with the ¨intelligent arm¨system was done by the Ethernet through router between a computer (where Galil software run to manually control debugging process and input object parameters), Galil motion controllers (FG and VG).

4.2 Software

Software implementation was done by using different programming environments and languages in Linux-based (Ubuntu) operation system. Languages, which were used are Galil programming language and C programming language. One of the development environment is Galil's freeware terminal module of GalilTools. It was used to write and debug low-level program modules on Galil language for Galil motion controller microprocessor. Also in freeware version of GalilTools ¨Watch¨module is available. Those two tools were completely enough for program development and testing. ¨Watch¨module was used to read switches and I/O signal conditions in the real-time. Screenshot of GalilTools is shown on figure 4.4.

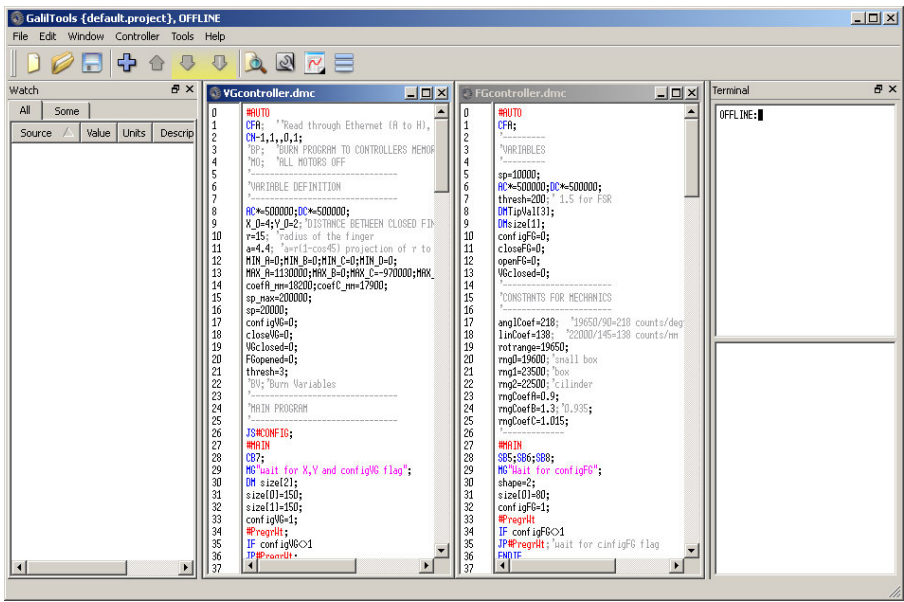

Figure 4.4: GalilTools programming and real-time monitoring environment

High-level user application requires libGalil.so library to communicate with Galil controller without GalilTools in Linux OS, libGalil.so library also required.

The implemented software modules are following:

1. flexible gripper (FG) module (Galil programming language)
2. vice gripper (VG) module (Galil programming language)
3. tactile sensors reading module to communicate with Weiss DSACON 32-H controller (high-level programming language)

4.2.1 Flexible gripper control program

Flexible gripper low-level control program provides autonomous operation of flexible gripper system. It fully runs in Galil motion controller and does not need any additional computation resources. It only needs input values and a start flag to begin a program flow. Input values are shape flag and dimensions of the object. When controller is on, program begins automatically. It waits for parameters of the object and flag to start configuration. When both are received, all fingers go to open position, to estimate start point of the linear motion counting. Then program calculates positions of the fingers by using input data and mechanical constants. These constants were estimated empirically. Some of them reflect dependencies between motor encoders counts and distance of motion. Some of them are constants such as a threshold value of the encoder error. They are the following:

1. coefficients of linear (counts of encoder / mm) and rotational (counts / degree) motions
2. fingers maximum runaway length in encoder counts for different configurations of the gripper (small box, big box, cylinder)
3. predefined values of the angels for three different configurations
4. threshold value of encoder error; used to define a moment when motor should be stopped if linear motion of the finger cannot be continue (touched an object)

Program sets fingers into pregrasping positions accordingly to the object size. Pregrasping means that positions of the fingers are almost at the object size point, but it leaves a space to approach an object and do not touch it. Configuration of the fingers is also set accordingly to the data about a shape of the object. At this point configuration has been finished. Program waits for "Close" flag and when it is set, start to grasp. Program continue to close the fingers until all motor stopped. It can occur in two cases: 1) if value which represents pressure on fingertips override threshold or 2) if value of motor error exceed its threshold. When all three motors stopped, program sets corresponding software and hardware flags and keeps holding positions of fingers. Gripper needs "Open" flags in order to release an object. After this flag is set, program also

sets another flag which can be used to control an external device or provide an information about the state of the gripper. Then program waits again for “Close” flag and if it is not set, checks for “configuration” flag, which means that a new object information has been obtained. Process starts over from the beginning.

Interesting aspect of this program is “hard” stop of grasping, or in other words “hardware sensitive grasping”. As it was mentioned above motor errors were used for this method. Error calculation in the controller is very accurate and one-to-one dependence between the increase in error and pressure on the fingertips can be estimated. A main shortcoming of this method is that with a predefined value of the error, increase in error does not reflect the increase in pressure. Also overloading of the controller can occur.
In FG control program in this phase of development there is only one mode of ˋsensitive grasping´- ˋhard grasping´which uses only motor errors.
Full flowchart of the FG control program is shown on figure 4.5.

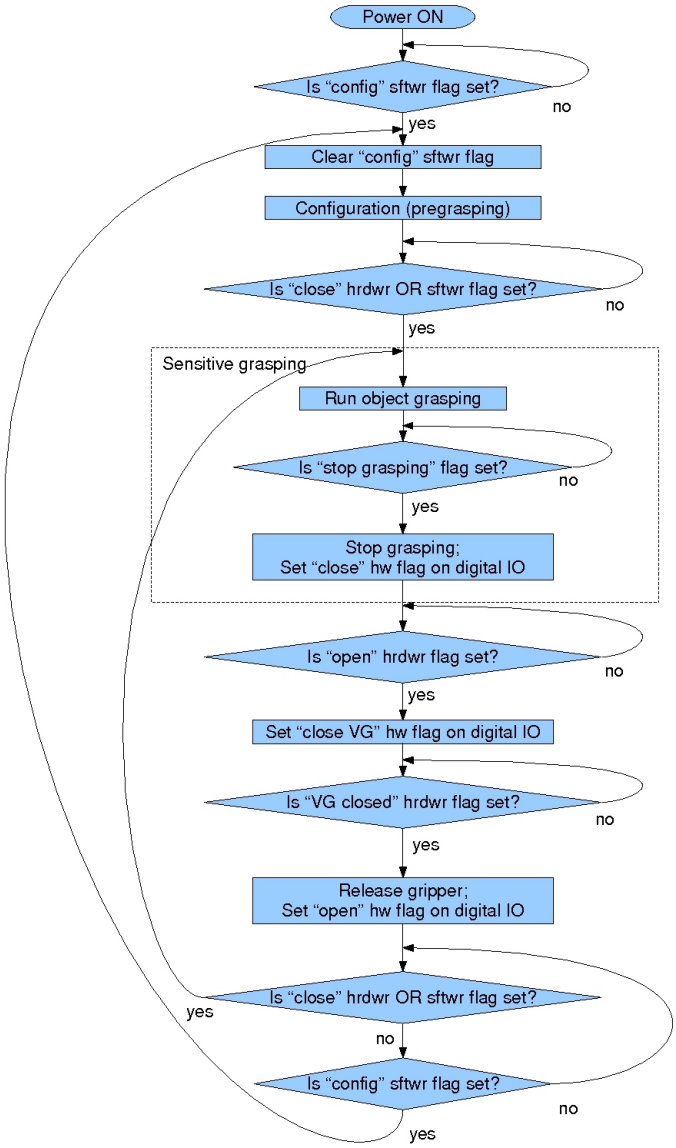

Figure 4.5: Flowchart of FG Galil control program

Flexible gripper control system connected with the ¨intelligent arm ˝controller via digital interface. As shown on a flowchart, communication with IRC5 ABB controller is done by system of flags and acknowledgements. All flags are duplicated in hardware and software manner. It provides a control possibility by using both hardware I/O interface and high-level software commands from ex-

ternal computer.

Fingers configuration mode depends on a shape of the object to be grasped. The information about an object is stored into the controller memory until the controller is turned on or new values arrive and overwrite an old information. There are three configuration modes: ˜small box˝for rectangular objects with max 200 mm in longest side, ˜box˝and ˜cylinder˝.

4.2.2 Vice gripper Galil control program

Input values for VG control program are also shape and dimensions of the object. There are only two possibilities for the object shapes: 1) square box or cylindrical object and 2) rectangular object. The decision about object's shape is made based on object dimensions. If X and Y length are equal, the it is an object from the first group (square box or cylinder). Otherwise object has a rectangular shape. Calculations of pregrasping position of the fingers are made based on mechanical dimensions of them and shown on figure 4.6.

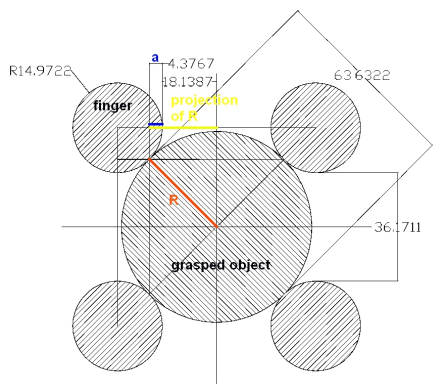

Figure 4.6: Calculations of VG fingers position

Therefore the actual position L of closed fingertips for square box or cylinder can be calculated as follows:

$$L = \frac{R \cos 45}{2} + a$$

Flowchart of VG Galil control program is shown on figure 4.7.

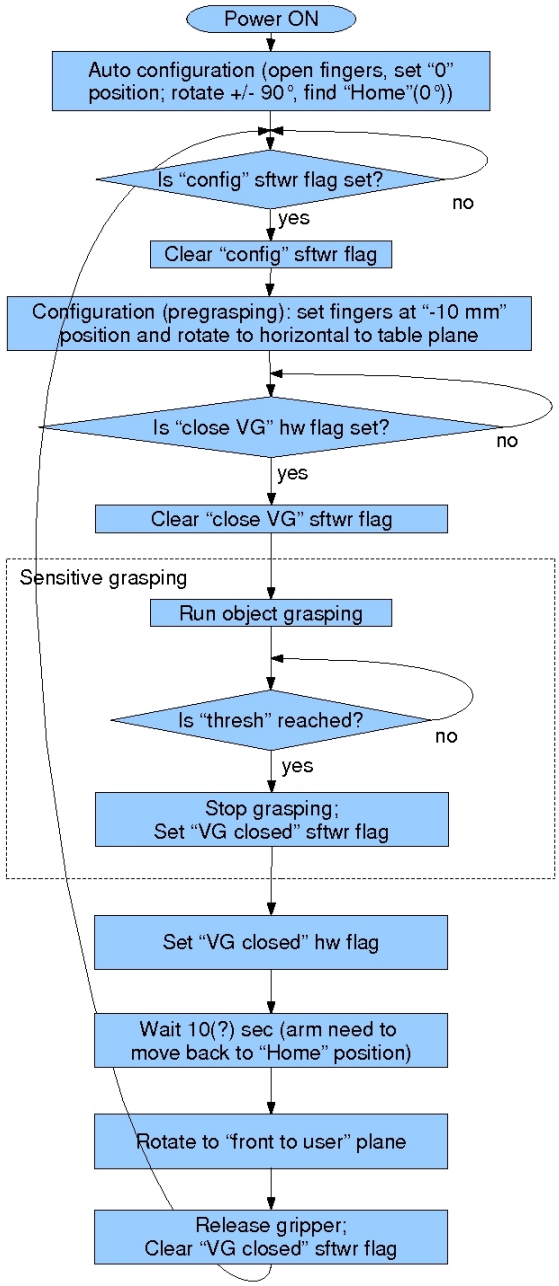
Power ON
Auto configuration (open fingers, set "0" position; rotate +/- 90°, find "Home"(0°))
Is "config" sftwr flag set?
no
yes
Clear "config" sftwr flag
Configuration (pregrasping): set fingers at "-10 mm" position and rotate to horizontal to table plane
Is "close VG" hw flag set?
no
yes
Clear "close VG" sftwr flag
Sensitive grasping
Run object grasping
Is "thresh" reached?
no
yes
Stop grasping;
Set "VG closed" sftwr flag
Set "VG closed" hw flag
Wait 10(?) sec (arm need to move back to "Home" position)
Rotate to "front to user" plane
Release gripper;
Clear "VG closed" sftwr flag

Figure 4.7: Flowchart of VG Galil control program

4.2.3 Tactile sensors system module

Weiss Robotics DSACON32-H controller uses a specific communication protocol to acquire values of the pressure from sensors. It also can be used to adjust the controller parameters. Protocol description were used to write a required programming module for the tactile data acquisition.

General communication protocol Regardless of the used interface, the sensor controller communicates with its host via binary data packets. They consist of a preamble signalling of the beginning of a new data packet. An identification byte describes the content of the packet. It is used to distinguish the several commands of the controller. The two byte size value determines the size of the packet's payload in bytes. For simple signalling packets without any payload, this value is 0, and the following payload and the checksum are omitted. If the packet contains data, a two byte CRC16 checksum is added. The polynom is 0x1021.

To send a command to the sensor controller, the packet ID has to be set according to the command ID. An ID of 0x00 identifies current measurement values. Such messages are sent exclusively by the sensor controller.

Format of general protocol message is shown in the table on figure 4.8.

Byte	Symbol	Description
0..2	PREAMBLE	Signals the begin of a new message and has to be 0xAAAAAA
3	PACKET_ID	ID byte of the packet.
4..5	SIZE	Size of the packet's payload in bytes. May be 0 for signalling packets.
6..n	PAYLOAD	Payload data
n+1	CHECKSUM	CRC16 checksum of the whole data packet. The polynom is 0x1021 (CCITT-16). **Please note:** The checksum is only transferred with packets containing a payload. On signalling packets, it is omited.

Figure 4.8: Format of DSACON32-H communication protocol

Example 1: Signalling packet:

$$< 0xAA >< 0xAA >< 0xAA >< 0x01 >< 0x00 >< 0x00 >$$

Example 2: Packet with payload:

$$< 0xAA >< 0xAA >< 0xAA >< 0x01 >< 0x02 >< 0x00 >< 0xCD >$$

$$< 0xAB >< 0x?? >< 0x?? >$$

Format of data acquired When the data acquisition is requested, the controller sends data packets containing the acquired data. The ID of a data packet is always 0x00. The payload contains a four byte timestamp which reflects the sensor controller time in milliseconds when the frame was acquired. Therefore this can be used to reconstruct the chronological sequence of the data.

Format of data acquisition message (controller respond) is shown in the table on figure 4.9.

Byte	Symbol		Description
0..2	PREAMBLE		Signals the begin of a new message and has to be 0xAAAAAA
3	PACKET_ID		ID byte of the packet = 0x00
4..5	SIZE		Size of the packet's payload in bytes.
6..9	PAYLOAD	TIMESTAMP	Sensor controller time in milliseconds when the passed frame was acquired
10		FLAGS	D7 1: RLE compression is activated 0: No compression D6..0 reserved
11..n		FRAME _DATA	Raw or compressed frame data
n+1	CHECKSUM		CRC16 checksum of the whole data packet. The polynom is 0x8408.

Figure 4.9: Format of data acquisition protocol (respond from controller)

Example of uncompressed data frame of a sensor with 16 texels, sampled at processor time 8197 is shown in the table on figure 4.10.

Data	Description
0xAA 0xAA 0xAA	Preamble
0x00	Packet ID
0x14 0x00	Size of the payload is 20 bytes
0x05 0x20 0x00 0x00	Time in milliseconds since starting the sensor controller when the frame was acquired
0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x04 0xFF 0x00 0x00 0x00 0x00 0x00 0x00 0x12 0x1A 0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x00	Raw frame data, starting with the upper left texel of the measurement converter and going on line by line.
0x?? 0x??	Checksum

Figure 4.10: Data acquisition from 16 sensors example

Developed software module sends a data request to the controller. Then reads a controller request and if connection is established, reads data from three matrices of tactile sensors. Each time the data is received, it needs to be interpreted according to the protocol format. The interpretation consists of a transformation of the serial bytes of data into 12bit resolution decimal numbers. There are two possibilities of data interpretation available: 1) data delivered in format of one maximum value from each matrix, i.e. three values simultaneously and 2) data delivered in format of one maximum value through all three matrices.

4.2.4 Main control program

Main controller program should have been consisted of a few software elements to implement a high-level control program of both FG and VG grasping systems. One of them is a virtual server for SAND 92 computer, which is connected to the motion capture system and provides the object information data. Second is a sensitive grasping module, which uses tactile sensor system software module to provide a precise grasping with a given and adjustable force while carrying the object. And third one is an object information data translator, which converts data from the motion capture system to Galil control programs format.

Only some of them were actually accomplished in a form of separate modules. Virtual server module and tactile sensor system module are completed and ready to be used in further work. Some work is also done in high-level communication with Galil controller from user software without a special Galil

software module.

Chapter 5
Experimental Results

In order to demonstrate the thesis results, an Integrated test platform was built in the AASS research lab. The aims of the Integrated test platform are testing and debugging a gripper control program and adjustment of the integration of gripper into the "intelligent arm" system.

5.1 Integrated test platform content

The test platform consists of the following modules:

1. WEISS Robotics DSACON32-H tactile sensors controller and three DSA 9205 matrices of tactile sensors
2. Flexible Gripper (FG) Galil's controller
3. Vice Gripper (VG) Galil's controller
4. LAN router (or switch)
5. IRC5 ABB arm controller and IRB 140 ABB robotic arm
6. computer, connected to FG, VG and IRC5 via router. It is used to control a flow of both Galil based programs by using GalilTools and editing and uploading to IRC5 control scripts for the robotic arm

Schematic diagram of Integrated test platform is shown on figure 5.1

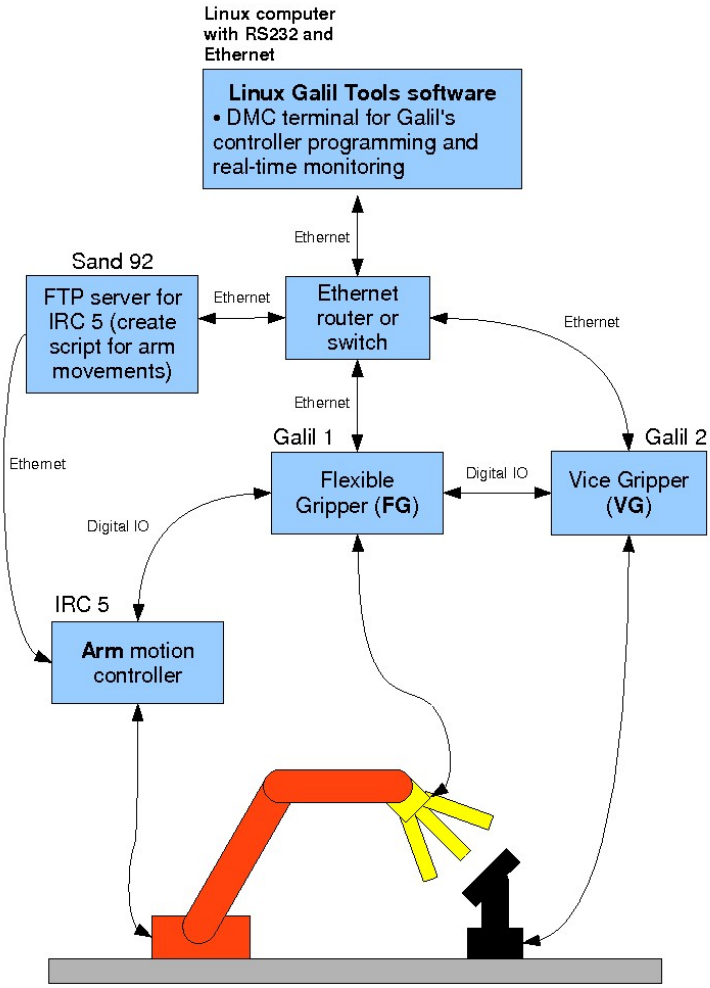

Figure 5.1: Functional elements of motion control system

Photos of the real Integrated test platform are shown on figures 5.2, 5.3 and 5.4.

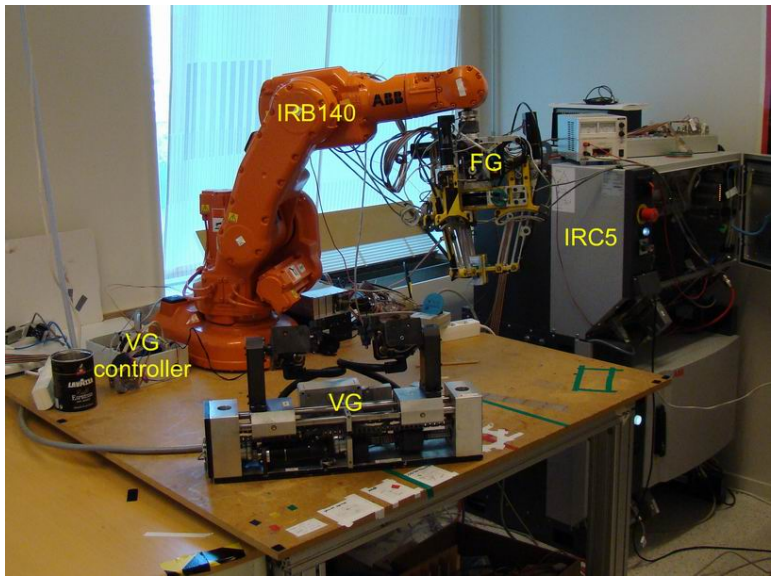

Figure 5.2: Integrated test platform picture 1

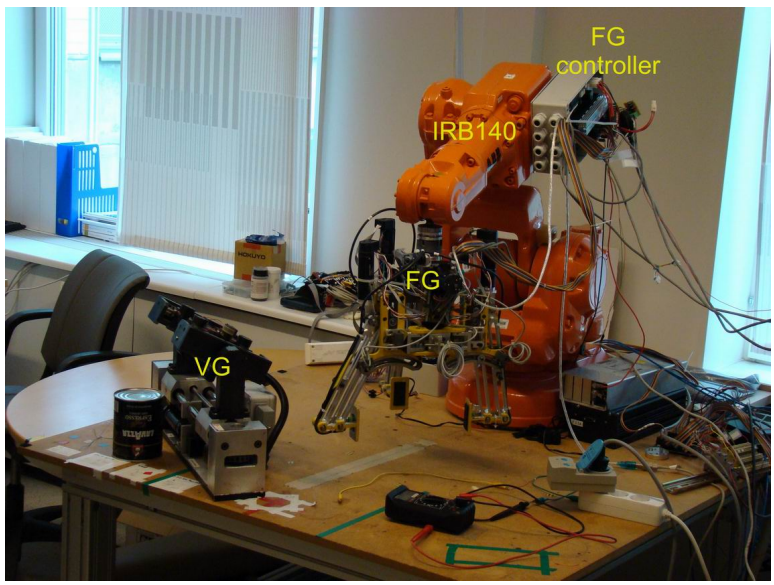

Figure 5.3: Integrated test platform picture 2

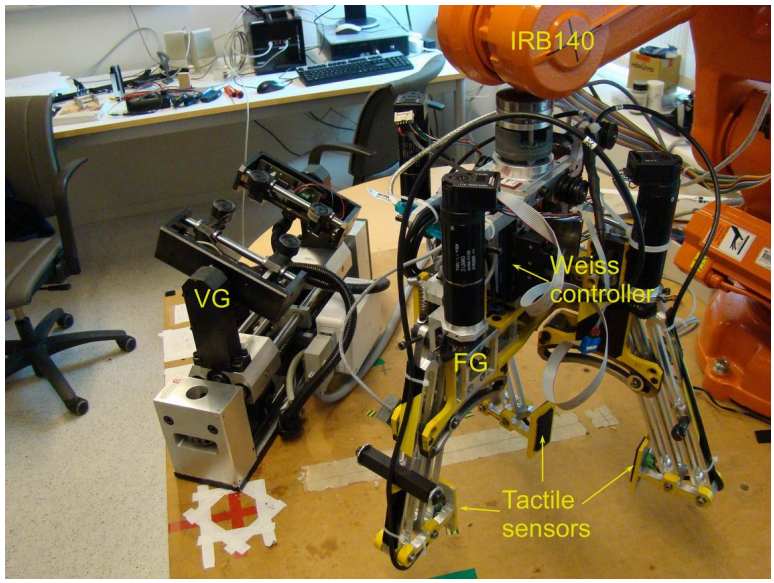

Figure 5.4: Integrated test platform picture 3

5.1.1 Integrated test platform description

The purpose of the Integrated test platform is to demonstrate and describe work of following six parts of the project:

1. flexible gripper (FG) control system
2. vice gripper (VG) control system
3. autonomous real-time communication between first Galil's systems (for FG) and IRC5 arm controller

Scenario of the experiment on Integrated test platform consists of the following steps:

1. Operator uploads an arm movements script into IRC5 and executes it. Then operator sets an object description to the FG and VG controllers from the computer via GalilTools software.
2. FG runs self configuration (open-close procedure in purpose to estimate finger's start position). Then FG sets fingers into the pregrasping position (fingers are in about 5 to 10 mm away from the object). FG waits for Grasp flag from IRC5.

3. While FG performs movements described in the previous section, arm moves from idle position (as shown on figure 5.2) towards the predefined position of the object. FG's self configuration and pregrasping runs about 15 seconds and this delay is added to the arm script when it reaches the object point (first point). When a waiting time expired IRC5 sets a grasp flag and after that FG grabs the object.

4. When FG finished grabbing the object, it sets a grasp acknowledgement flag and arm starts moving toward its second point at the VG (about 15 to 20 cm above the VG fingers). VG is already self configured according to the object description and ready to grasp an object.

5. Arm moves into VG in the direction perpendicular to the VG's fingers plane. This is the third point when FG's fingers are about 4 to 5 cm above the VG's fingers (to provide a secure grasping of the object by VG).

6. When arm is in the third point, IRC5 sets Open flag. Prior to release FG sets VGopen flag. When VG has grasped the object, it sets an acknowledgement. Only after that FG releases and sets an Open acknowledgement flag. The arm waits for this flag to continue to execute the script.

7. Arm moves into the fourth point (the same position as in second point - about 20 cm above the VG) at the same manner as from the second to the third points. VG has a delay after it grasps the object about 20 second. It is done to let the arm move away from the VG.

8. At the next step arm moves into its final, the fifth point which is idle position. VG moves (rotates) away from the arm and releases the object (this movement is similar to placing an object on a conveyor belt). Then VG opens fingers and rotates back into the idle position toward the arm (see figure 5.3) and waits for the new object description and/or VGclose flag.

Experiment flowchart is shown on figure 5.5.

Information about the position and orientation of the arm and all objects in the arm's working area can be taken from the cameras of the motion capture system installed above the Arm. But since motion control of the Arm is not a part of this project, the Arm can be controlled only manually to create a script of predefined step-by-step movements. Therefore, an Integrated test platform runs in supervision semi-autonomous mode. Orientation of the objects grabbed by FG with respect to the arm and flexible gripper are also specified. However, FG controller uses an information about the shape and size of the objects. Both FG and VG Galil controller run in autonomous mode. They communicate to IRC5 controller and to each other by using digital interface. Both of the programs are based on the steps of moves in the script for ABB arm. FG Galil controller can

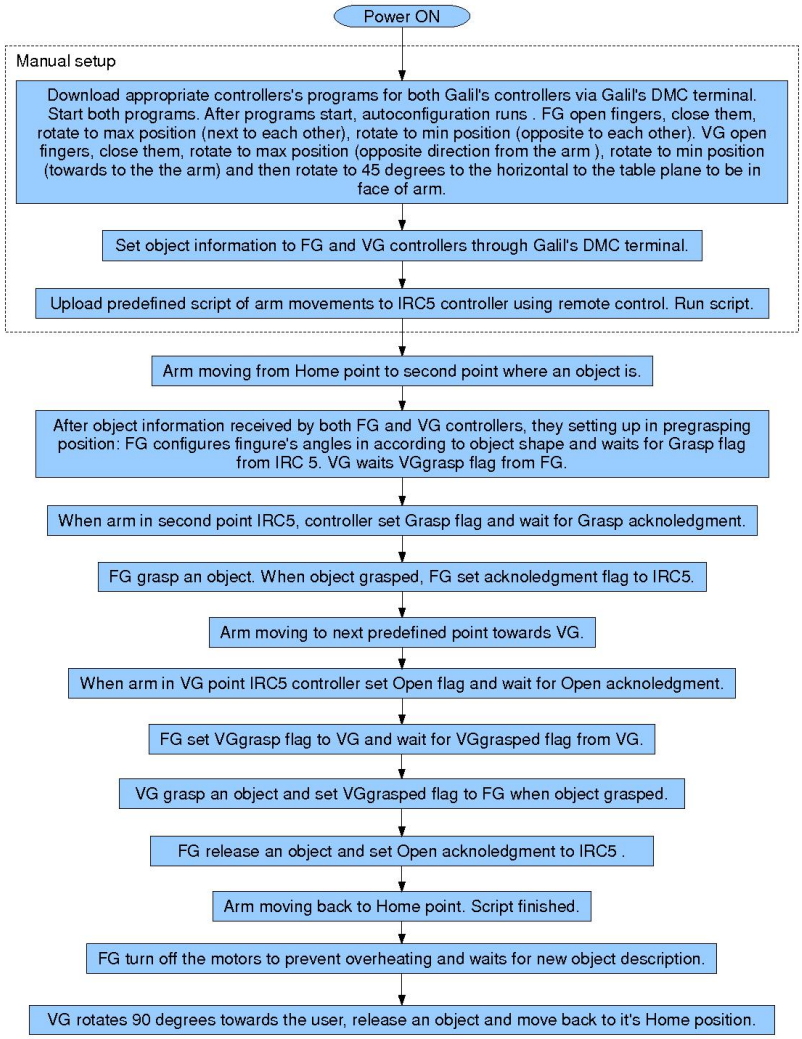
Power ON
Manual setup
Download appropriate controllers's programs for both Galil's controllers via Galil's DMC terminal. Start both programs. After programs start, autoconfiguration runs . FG open fingers, close them, rotate to max position (next to each other), rotate to min position (opposite to each other). VG open fingers, close them, rotate to max position (opposite direction from the arm), rotate to min position (towards to the the arm) and then rotate to 45 degrees to the horizontal to the table plane to be in face of arm.
Set object information to FG and VG controllers through Galil's DMC terminal.
Upload predefined script of arm movements to IRC5 controller using remote control. Run script.
Arm moving from Home point to second point where an object is.
After object information received by both FG and VG controllers, they setting up in pregrasping position: FG configures fingure's angles in according to object shape and waits for Grasp flag from IRC 5. VG waits VGgrasp flag from FG.
When arm in second point IRC5, controller set Grasp flag and wait for Grasp acknoledgment.
FG grasp an object. When object grasped, FG set acknoledgment flag to IRC5.
Arm moving to next predefined point towards VG.
When arm in VG point IRC5 controller set Open flag and wait for Open acknoledgment.
FG set VGgrasp flag to VG and wait for VGgrasped flag from VG.
VG grasp an object and set VGgrasped flag to FG when object grasped.
FG release an object and set Open acknoledgment to IRC5 .
Arm moving back to Home point. Script finished.
FG turn off the motors to prevent overheating and waits for new object description.
VG rotates 90 degrees towards the user, release an object and move back to it's Home position.

Figure 5.5: Flowchart of the Integrated test platform.

be run without VG controller. FG controller communicates with IRC5 by using already implemented protocol from the KTHand project. It uses open and close commands only. Both commands need acknowledgment, therefore program flow is self-monitored and the next step of the script can not be executed before the previous is finished.
In order to present the working system a video file has been made. The video file shows the flexible gripper integrated into the "intelligent arm". FG works together with IRC5 controller and Vice gripper. Therefore it is possible to say that the aims of this master's project were successfully achieved.

Chapter 6
Conclusion

In this thesis I achieved goals such as:

- developed a control program for the Flexible gripper prototype
- developed a control program for the Vice gripper prototype
- developed a tactile sensors communication and control software module
- integrated FG and VG into the existing “intelligent arm” system by means of digital interfacing.
- designed and implemented the Flexible gripper hardware wiring
- implemented a wiring for communication between IRC5, FG and VG

During the work on this thesis I learned the following:

- programming by demonstration technique
- Galil DMC2143 controller hardware
- Galil programming language
- software implementation of the serial communication protocol, data checksum calculation and data protocol of WEISS DSACON-32 tactile sensors controller
- work with the “real hardware” devices such as ABB IRB140 robotic arm and ABB IRC5 arm controller
- implementation of the electrical circuits

During the work on this thesis I faced with some challenges. One of them when was tactile sensors communication and control system. It took me a lot of time and efforts to implement a given data interface protocol and acquired data translation into my own application.

Another big challenges were the wiring of the FG and first run of the Flexible gripper prototype. There were some mistakes in wiring, some mechanical related problems and signals interference issues.

The future work is to develop a high-level software application which will combine features such as:

- reading of an object description from the Motion Capture system
- translation of the object description into a specific format for FG and VG control systems
- uploading the object description into FG and VG controllers
- combining all these new features with the already existing tactile sensors module to achieve the “precise sensitive grasping”.

Bibliography

[1] L. Biagiotti, C. Melchiorri, G. Vassura, *A dexterous robotic gripper for autonomous grasping*, Industrial Robot: An International Journal, Vol 30, November 5, 2003, pp 449 - 458.

[2] S. Hesse, *Grippers and their applications*, Ruiter Straße 82, 2004, p 19.

[3] A. Wolf, R. Steinmann, H. Schunk, *Grippers in Motion*, Springer-Verlag Berlin Heidelberg, 2005, p 41.

[4] A. Wolf, R. Steinmann, H. Schunk, *Grippers in Motion*, Springer-Verlag Berlin Heidelberg, 2005, p 44.

[5] www.weiss-robotics.de

[6] www.weiss-robotics.de

[7] http://www.interlinkelectronics.com/force_sensors/technologies/fsr.html

[8] http://www.weiss-robotics.de

[9] http://www.weiss-robotics.de

[10] Galil Motion Control Company, www.galilmc.com

[11] http://www.galilmc.com/support/manuals/21x3_accessories.pdf, Chapter 9

[12] http://www.galilmc.com/support/manuals/21x3_accessories.pdf, Chapter 10

[13] http://www.galilmc.com/support/manuals/man21x2.pdf

[14] http://www.galilmc.com/support/manuals/man21x2.pdf, Chapter 10

[15] http://netbeans.org/

[16] A. Bicchi and V. Kumar, *Robotic grasping and contact: A review*, in Proceedings of the IEEE International Conference on Robotics and Automation, ICRA 00, pp. 348 - 353, 2000.

[17] A. Bicchi. *Hands for dexterous manipulation and robust grasping: A difficult road towards simplicity*. IEEE Trans. on Robotics and Automation, 16(6):652Ű662, Dec 2000.

[18] N. S. Pollard, *Closure and quality equivalence for efficient synthesis of grasps from examples*, International Journal of Robotic Research, vol. 23, no. 6, pp. 595-613, 2004.

[19] A. Morales, E. Chinellato, A. H. Fagg, and A. del Pobil, *Using experience for assessing grasp reliability*, International Journal of Humanoid Robotics, vol. 1, no. 4, pp. 671-691, 2004.

[20] R. Platt Jr, A. H. Fagg, and R. A. Grupen, *Extending fingertip grasping to whole body grasping*, in Proc. of the Intl. Conference on Robotics and Automation, 2003.

[21] A. T. Miller, S. Knoop, H. I. Christensen, and P.K. Allen. *Automatic grasp planning using shape primitives*. In In Proceedings of the IEEE International Conference on Robotics and Automation, pages 1824-1829, 2003.

[22] A. Skoglund, B. Iliev, and R. Palm. *A hand state approach to imitation with a next-state-planner for industrial manipulators*. In Int. Conf. on Cognitive Systems, Karlsruhe, Germany, April 2008.

[23] J. Tegin and J. Wikander. *Tactile sensing in intelligent robotic manipulation Ű a review*. Industrial Robot, 32(1):64-70, 2004.

[24] K. Charusta, D. Dimitrov, A. J. Lilienthal and B. Iliev, *Extraction of Grasp Related Features by Human Dual-Hand Object Exploration*. Proceedings of the IEEE International Conference on Advanced Robotics (ICAR), 2009, pp. 122-127.

[25] A. Saxena, J. Driemeyer, and A. Y. Ng, *Robotic grasping of novel objects using vision*, Int. J. Rob. Res., vol. 27, no. 2, pp. 157-173, 2008.

[26] T. Yoshikawa, M. Koeda, and H. Fujimoto, *Shape recognition and grasping by robotic hands with soft fingers and omnidirectional camera* , in Robotics and Automation, 2008. ICRA 2008. IEEE International Conference on, May 2008, pp. 299-304.

[27] A. Morales, P. J. Sanza, Angel, and A. H. Fagg, *Vision-based threefinger grasp synthesis constrained by hand geometry* , Robotics and Autonomous Systems, vol. 54, no. 6, pp. 496-512, 2006.

[28] K. Huebner and D. Kragic, *Selection of robot pre-grasps using boxbased shape approximation* , Intelligent Robots and Systems, 2008. IROS 2008. IEEE/RSJ International Conference on, pp. 1765-1770, Sept. 2008.

[29] K. Huebner, S. Ruthotto, and D. Kragic, *Minimum volume bounding box decomposition for shape approximation in robot grasping* , Robotics and Automation, 2008. ICRA 2008. IEEE International Conference on, pp. 1628-1633, May 2008.

[30] D. J. Sturman and D. Zeltzer, *A survey of glove-based input* , IEEE Computer Graphics and Application, 1994.

[31] M. Hueser, T. Baier, and J. Zhang, *Learning of demonstrated grasping skills by stereoscopic tracking of human head configuration* , Robotics and Automation, 2006. ICRA 2006. Proceedings 2006 IEEE International Conference on, pp. 2795-2800, May 2006.

[32] *Phasespace motion capture* . http://www.phasespace.com/

[33] A. Billard and R. Siegwart, *Robot learning from demonstration*, Robotics and Autonomous Systems, vol. 47, no. 2-3, pp. 65-67, 2004.

[34] A. Skoglund, T. Duckett, B. Iliev, A. J. Lilienthal, and R. Palm, *Teaching by Demonstration of Robotic Manipulators in Non-Stationary Environments*. Proceedings of the IEEE International Conference on Robotics and Automation (ICRA), 2006, pp. 4339-4341.

[35] J. Tegin, S. Ekvall, D. Kragic, J. Wikander, and B. Iliev. *Demonstration based learning and control for automatic grasping*. In Intelligent Service Robotics, 2008.

[36] A. Skoglund, J. Tegin, B. Iliev, and R. Palm, *Programming-by-Demonstration of Reaching Motions for Robot Grasping*. Proceedings of the 2009 14th International Conference on Advanced Robotics, 2009

[37] R. Dillmann, *Teaching and learning of robot tasks via observation of human performance*, Robotics and Autonomous Systems, vol. 47, no. 2-3, pp. 109-116, 2004.

[38] A. Skoglund, B. Iliev, B. Kadmiry, and R. Palm, (*Programming by demonstration of pick-and-place tasks for industrial manipulators using task primitives, in IEEE International Symposium on Computational Intelligence in Robotics and Automation, Jacksonville, Florida, June 20-23 2007, pp. 368Ű373.*